

Management virtualizace

Bakalářská práce

Lukáš Komárek

Vedoucí bakalářské práce: Ing. Václav Novák, CSc.

Jihočeská univerzita v Českých Budějovicích

Pedagogická fakulta

Katedra informatiky

2013

bude připojen k internetu, proto musíte vyplnit, jaká fyzická síťová karta bude připojena k internetu. Ve spuštěném routeru zadat výchozí bránu v síti 10.10.13.1 a zprovoznit routování pro ostatní počítače v síti - tedy ty, jenž jsou ve scriptu označeni jako klient1-4.

Dovedu si představit použití tohoto scriptu při cvičení třeba na nastavování Active Directory či na spojování různorodých systémů pomocí Samby.

1.2.7 Koncem to nekončí

Kapitolou se scripty bych mé tipy ukončil, ovšem to neznámá, že KVM nemá více možností. Tento úžasný virtualizační nástroj má parametry, o kterých jsem se zde nezmiňoval. Zkuste použít manuálové stránky a něco se přiučit z nich. Už jste slyšeli o parametru -nographic a VNC připojení?

Prohlášení

Prohlašuji, že v souladu s § 47b zákona č. 111/1998 Sb. v platném znění souhlasím se zveřejněním své bakalářské práce, a to v nezkrácené podobě fakultou elektronickou cestou ve veřejně přístupné části databáze STAG provozované Jihočeskou univerzitou v Českých Budějovicích na jejích internetových stránkách, a to se zachováním mého autorského práva k odevzdanému textu této kvalifikační práce. Souhlasím dále s tím, aby toutéž elektronickou cestou byly v souladu s uvedeným ustanovením zákona č. 111/1998 Sb. zveřejněny posudky školitele a oponentů práce i záznam o průběhu a výsledku obhajoby kvalifikační práce. Rovněž souhlasím s porovnáním textu mé kvalifikační práce s databází kvalifikačních prací Theses.cz provozovanou Národním registrem vysokoškolských kvalifikačních prací a systémem na odhalování plagiátů.

Datum

Podpis studenta

```

tunctl -t vnic0
ip a add 10.10.13.1/24 brd + dev vnic0
ip link set up dev vnic0
iptables -t nat -A POSTROUTING -o $nic -j MASQUERADE
echo 1 > /proc/sys/net/ipv4/ip_forward

#Spouštění routeru
kvm -hda $cesta$router -m ${router_ram}M -snapshot \
-net nic,macaddr=52:54:00:12:12:12 \
-net tap,ifname=vnic0,script=no \
-net nic,macaddr=52:54:00:13:bb:13,vlan=1 \
-net socket,listen=:1234,vlan=1&

#spouštění klientů
for obraz in $klient1 $klient2 do kvm -hda $cesta$obraz -m ${moc_ram}M -snaps
-net nic,macaddr=52:54:00:33:22:${RANDOM:0:1}${RANDOM:0:1} \
-net socket,connect=127.0.0.1:1234&
done

for obraz in $klient3 $klient4
do
kvm -hda $cesta$obraz -m ${malo_ram}M -snapshot \
-net nic,macaddr=52:54:00:33:28:${RANDOM:0:1}${RANDOM:0:1} \
-net socket,connect=127.0.0.1:1234&
done

```

Může se stát, že budete potřebovat spouštět různorodé systémy, které mají jiné nároky. Proto jsem vám připravil script, kterým je možno spustit dva druhy operačních systémů a každému typu přidělit jinou velikost RAM. Nemusíte se omezovat jen na paměti, script se nechá jednoduše upravit i na jiné rozdílné parametry - je to pouze na vás. Stejně jako u předchozího scriptu, i zde doplníte názvy souborů atp. Ve scriptu se spouští i jeden router, který

ZADÁNÍ BAKALÁŘSKÉ PRÁCE
(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: **Lukáš KOMÁREK**
Osobní číslo: **P08252**
Studijní program: **B7507 Specializace v pedagogice**
Studijní obor: **Informační technologie ve vzdělávání**
Název tématu: **Management virtualizace**
Zadávající katedra: **Katedra informatiky**

Z á s a d y p r o v y p r a c o v á n í :

Optimalizace virtualizace výpočetního výkonu se stále diskutuje na stránkách odborných časopisů a zejména na internetových stránkách. Podstatně důležitější ale je managing virtualizace. Student představí systémy řízení virtualizace a virtualizaci samu. Zhodnotí vhodnost virtualizačního software napříč operačními systémy. Připraví optimální architekturu virtuálních systémů vhodných pro různé typy škol (počítačové učebny a laboratoře na základních, středních a vysokých školách), jak pro výukové účely, tak pro zabezpečení správy virtuálních systémů. Zadání předpokládá návrh a otestování různého stupně virtualizace, od hardwarové až po aplikační, doplněné o testy navržených technologií.

Script není problém předělat na spouštění více počítačů pouhým zavedením proměnné s názvem virtuálního disku a tuto proměnnou dopsat do cyklu for.

V případě nevyplnění názvu virtuálního disku se nic špatného nestane, prostě se jeden virtuální počítač nespustí.

Script na připojení sítě k internetu, různé typy OS

```
#!/bin/bash

#cesta k síťovému disku
sitovy_disk=/media/nfs/

#název souboru routeru a jeho RAM v MB
router=debian.img
router_ram=128

#1. skupina klientů a jejich RAM v MB
klient1=windows7.img
klient2=windowsXP.img
moc_ram=768

#2. skupina klientů a jejich RAM v MB
klient3=xubuntu.img
klient4=debian.img
malo_ram=256

#Síťová karta v internetu
nic=eth0

#Nastavení sítě
```

Rozsah grafických prací:

Rozsah pracovní zprávy:

60

Forma zpracování bakalářské práce: **tištěná**

Seznam odborné literatury:

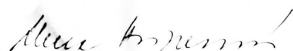
1. Microsoft [online]. Praha : Microsoft, 2010 [cit. 2010-04-19]. Virtualizace. Dostupné z WWW: <<http://www.microsoft.com/cze/virtualizace/>>.
2. CONNECT [online]. 2010 [cit. 2010-04-19]. Kategorie Virtualizace. Dostupné z WWW: <<http://connect.zive.cz/serverova-virtualizace>>.
3. ÚVT MU [online]. 13.4.2010 [cit. 2010-04-19]. Virtualizace. Dostupné z WWW: <<http://ics.muni.cz/zpravodaj/articles/545.html>>.
4. NetWin [online]. 13.4.2010 [cit. 2010-04-19]. Technický seminář: virtualizace desktopů, Citrix XenDesktop. Dostupné z WWW: <<http://www.netwin.cz/seminaregclid=COX8qIf0kqECFRIrDgodfXHhRA>>.

Vedoucí bakalářské práce:

Ing. Václav Novák, CSc.
Ústav aplikované informatiky

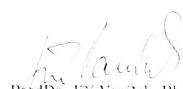
Datum zadání bakalářské práce: **19. dubna 2010**

Termín odevzdání bakalářské práce: **30. dubna 2011**



doc. PhDr. Alena Hošpesová, Ph.D.

děkanka



PaedDr. Jiří Vaniček, Ph.D.

vedoucí katedry

V Českých Budějovicích dne 19. dubna 2010

Automatické spuštění několika virtuálních strojů s náhodnou mac adresou:

```
#!/bin/bash

#cesta k síťovému disku
sitovy_disk=/media/nfs/

#názvy souborů s virtuálními počítači
prvni=centOS.img
druhy=gentoo.img
treti=openBSD.img

#spuštění počítačů
for obraz in $prvni $druhy $treti
do
kvm -hda $sitovy_disk$obraz -snapshot -m 256M \
-net nic,macaddr=52:54:00:54:${RANDOM:0:2}:${RANDOM:0:1}${RANDOM:0:1} \
-net socket,mcast=230.0.0.1:1234&
sleep 2s
done
```

Ve scriptu je potřeba doplnit cestu k síťovému disku a názvy souborů s virtuálními disky. Při spouštění scriptu rovnou ze síťového disku(= aktuální adresář bude v tomto případě /media/nfs/) je možné místo cesty zadat ./ atp.. Potom je vhodný k rozeslání zpráv na spuštění třech virtuálních počítačů na každém hostiteli. Pokud jsou hostitelské počítače v jedné síti, budou propojeny do sítě virtuální počítače ze všech hostitelů.

Náhodná mac adresa není ošetřena na 100%, v některých případech by se mohlo stát, že se vygeneruje stejná adresa. Vyloženě špatná adresa, kvůli které by se jeden virtuální počítač nespustil vůbec, se také může vygenerovat (kvůli \${RANDOM:0:2} - můžete vyměnit za dvě jedničky, jako je tomu vedle), ovšem tato pravděpodobnost je velmi nízká.

Abstrakt

Tato bakalářská práce se zabývá aktuální problematikou virtualizace. V mé práci dávám přednost otevřeným řešením, ovšem rozhodně se nechci vyhnout i jiným – rozšířeným možnostem virtualizace. Management virtualizace řeším dle možností daného systému (Linux/ KVM – Bash/ Scripty, VMware Workstation – uživatelské rozhraní, ESXi - vSphere Center, XenServer - XenCenter apod.) a nejlepším způsobem se budu věnovat více do hloubky a vytvořím souhrn tipů k výuce sítí podpořené tímto virtualizačním nástrojem a souhrn tipů pro virtualizovanou učebnu. U každého typu virtualizace se zajímám o nastavitelné parametry, vhodnost jednotlivých možností hlavně pro síťovou komunikaci mezi virtuálními počítači. Nevyhnu se ani porovnání formátů pro virtuální disky, jejich tvorbě, úpravě, formátování na konkrétní File systém, jaký přečte virtuální operační systém a celkovému zapojení disků do virtuálních systémů.

Abstract

This work deals question of management of virtualization. I prefer open solutions, but i accept other possibilites of virtualization. I deal with management of virtualization as the system allows (Linux/ KVM – bash/ scripts, VMware Workstation – user interface etc.). I will spend more time with better virtualization software. I will make summary of tips for teaching networking supported by virtualization and summary of tips for VDI. I'm interested in adjustable parameters, appropriateness solutions for networking virtual machines. I will compare formats of virtual disks. I will create and edit virtual disks and i will format disks to file systems for the operating system and global adding disks to virtual systems.

VM2:

```
ip a add 192.168.101.2/24 brd + dev eth0
ip link set up dev eth0
ip route add default via 192.168.101.1
echo "nameserver ip.adresa.dns.serveru" >> /etc/resolv.conf
# Druhé síťové kartě dáme adresu v síti s ostatními virtuálními
# počítači a nastavíme routování
ip a add 10.0.0.3/24 brd + dev eth1
ip link set up dev eth1
iptables -t nat -A POSTROUTING -o eth0 -j SNAT --to 192.168.101.2
echo "1" > /proc/sys/net/ipv4/ip_forward
```

Počítačům ve virtuální síti nastavíme jako výchozí bránu VM2, DNS a je hotovo.

1.2.6 KVM ve scriptech

To, že se KVM ovládá přes BASH, je možné využít pro spouštění mnoha počítačů a tvorby sítě pomocí scriptů. Tvorba scriptů je neomezená a nechá se udělat script v podstatě na vše. Zde vám předvedu pouze pár ukázek, které můžete využít pro urychlení vaší práce.

Poděkování

Rád bych poděkoval vedoucímu mé bakalářské práce,
panu Ing. Václavu Novákovi, CSc.
za profesionální přístup.

Řekněme, že hostitelský počítač je připojen k síti, případně k internetu, síťovou kartou eth0 s pevnou IP adresou 192.168.100.5/24. Nastavme na něm SNAT a aktivujme IP forwarding:

```
iptables -t nat -A POSTROUTING -o eth0 -j SNAT --to 192.168.100.5
echo "1" > /proc/sys/net/ipv4/ip_forward
```

Vytvoříme virtuální síťové rozhraní vnic0 a přiřadíme mu IP adresu z jiné sítě:

```
tunctl -t vnic0
ip a add 192.168.101.1/24 brd + dev vnic0
ip link set up dev vnic0
```

Nyní spustíme virtuální počítač VM2 a dáme mu dvě síťové karty. Jednu jeho síťovou kartu připojíme k virtuální “kartě” vnic0, druhou připravíme na spojení s virtuální sítí tvořenou multicastem:

```
kvm -hda virtual.img -m 256M -snapshot \
-net nic,macaddr=52:54:00:12:12:12 \
-net tap,ifname=vnic0,script=no \
-net nic,macaddr=52:54:00:13:13:13,vlan=1 \
-net socket,mcast=230.0.0.1:1234,vlan=1
```

Nyní nastavíme na VM2 IP adresu v síti s vnic0, výchozí brána bude nastavena na IP adresu virtuálního rozhraní hostitelského počítače. Pokud budeme chtít přistupovat k internetu, zadáme i DNS servery (musí být ,v našem případě, připojen hostitelský počítač síťovou kartou eth0):

Obsah

I	Úvod	10
1	Cíle práce	11
II	Teoretická část	12
2	Virtualizace na desktopech	12
2.1	Operační systém	13
2.1.1	Linux	13
2.1.2	Windows	14
2.2	Virtualizační nástroje	15
2.2.1	KVM	15
2.2.2	VMware Workstation	16
2.2.3	Windows Virtual PC	17
3	Nativní virtualizace	17
3.1	Virtualizační nástroje	18
3.1.1	VMware ESXi	18
3.1.2	CITRIX XenServer	19
4	Jiné případy virtualizace	19
4.0.3	Cloud computing	19
5	Výuka sítí podpořená virtualizací	20
5.1	Výběr operačního systému	21
5.2	Výběr virtualizačního nástroje	22
5.3	Pro koho je souhrn praktických tipů určen	23
5.4	Co je a není cílem sepsaného souhrnu tipů	23

VM1:

```
ip a add 10.0.0.2/24 brd + dev eth0  
ip link set up dev eth0
```

VM0:

```
ping 10.0.0.2
```

VM1:

```
ping 10.0.0.1
```

Nastavování IP adres virtuálním počítačům můžete využít jako první úkol v hodině sítě podpořené virtualizací.

1.2.5 Připojení sítě virtuálních počítačů do skutečné sítě

Pro případ, že budeme chtít aby virtuální počítače komunikovaly s počítači mimo virtuální síť, využijeme hostitelský počítač jako router. Skutečnou síť, do které je připojen hostitelský pc skryjeme za NAT a na onom PC vytvoříme virtuální síťové rozhraní, ke kterému bude připojen minimálně jeden virtuální počítač. Na hostitelském počítači musí být nainstalován balík `uml-utilities`.

Dle předchozího návodu máme vytvořenou síť virtuálních počítačů, kterou chceme připojit do skutečné sítě. Proto spustíme jeden virtuální počítač, který bude zastávat funkci routeru. Tento virtuální počítač bude připojen jedním síťovým rozhraním ke skutečné síti a druhým do virtuální sítě.

6	Vyrtualizovaná učebna	24
6.1	Softwarové vybavení serveru	24
6.2	Pro koho je souhrn tipů pro VDI určen	24
6.3	Co je a není cílem sepsaného souhrnu tipů	24
III	Praktická část	26
7	Cíle	26
8	Testování virtualizačního softwaru pro desktopovou virtualizaci	26
8.1	Metodika	26
8.2	VMware Workstation 27	
8.3	KVM 30	
8.4	Windows Virtual PC	34
8.5	Výběr nástroje	35
9	Testování virtualizačních nástrojů pro nativní virtualizaci	36
9.1	Metodika	36
9.2	VMware ESXi	36
9.3	CITRIX XenServer	38
9.4	VDI-in-a-Box	40
9.5	Výběr hypervizoru	42
10	Tvorba souhrnu praktických tipů nejen pro výuku sítí podpořenou virtualizací	43
10.1	Rozbor kapitol souhrnu praktických tipů pro výuku sítí podpořenou virtualizací	43
10.1.1	Popis problému 43	
10.1.2	Hardwarové vybavení počítače 43	

Pomocí multicastu Každému počítači dáme síťovou kartu naslouchající stejnému UDP multicastu.

Spuštění prvního počítače (VM0):

```
kvm -hda linux.img -snapshot -m 256M \  
-net nic,macaddr=52:54:00:12:34:56 \  
-net socket,mcast=230.0.0.1:1234
```

Spuštění dalšího počítače (VM1):

```
kvm -hda linux.img -snapshot -m 256M \  
-net nic,macaddr=52:54:00:65:43:21 \  
-net socket,mcast=230.0.0.1:1234
```

Oba příkazy jsou v podstatě stejné, jen s rozdílnou mac adresou. Pomocí `-net socket,mcast=...` řekneme síťové kartě, k jaké adrese a portu multicastu se má připojit.

Funkční síť Ať už jste si vybrali jakoukoli možnost spojení počítačů do virtuální sítě, na spuštěných virtuálních počítačích nastavte síťovým kartám IP adresy ve stejné síti a přesvědčte se, zda na sebe vidí. Jistě říkám zbytečně, že do příkazových řádek virtuálních počítačů budete psát něco jako toto:

VM0:

```
ip a add 10.0.0.1/24 brd + dev eth0  
ip link set up dev eth0
```

10.1.3	Softwarové vybavení počítače	
	44	
10.1.4	Instalace virtuálního operačního systému	
	45	
10.1.5	Připojení sítě virtuálních počítačů do skutečné sítě	
	49	
10.1.6	KVM ve scriptech	50
10.1.7	Koncem to nekončí	50
10.2	Souhrn praktických tipů pro tvorbu virtuální desktopové in-	
	frastruktury	50
10.2.1	Virtuální desktopová infrastruktura	50
10.2.2	Instalace serveru	50
10.2.3	Ovládání serveru	50
10.2.4	VDI-in-a-box	51
11	Závěr	52
11.1	Použité zdroje	53
11.2	Přílohy práce	55

Další stroj tímto příkazem (VM1)

```
kvm -hda linux.img -m 256M -snapshot \  
-net nic,macaddr=52:54:00:33:22:11,model=virtio\  
-net socket,connect=127.0.0.1:1234
```

Parametr `-snapshot` zde je kvůli tomu, že spouštíme počítače ze stejného virtuálního disku. Pokud budeme spouštět najednou rozdílné počítače, můžeme `-snapshot` vynechat.

`-net nic` nastavuje síťovou kartu, se kterou bude virtuální počítač spuštěn, v tomto případě nám stačí nastavit jen její mac adresu. Model `virtio` nastavuji proto, že je s ní rychlejší datový přenos, než s emulovanou kartou `RTL8139`, která by byla ve virtuálním počítači, kdybychom model nezadali. U některých operačních systémů je potřeba doinstalovat kartě `virtio` ovladače - odkaz ke stažení je na oficiálních stránkách KVM. Při spouštění více virtuálních počítačů dbejte na to, aby dva počítače neměly stejnou mac adresu.

U prvního počítače nesmíme zapomenout nastavit port, na kterém bude naslouchat síťovému provozu, zde `1234`. Další počítače spouštíme obdobně, ale necháme je připojovat přes `loopback` fyzického počítače na stejný port, jako jsme nastavili u prvního stroje. `Loopback` je většinou nastaven na `127.0.0.1`, případně se nechá zjistit programem `ip` - síťové rozhraní `lo` nebo napsáním příkazu

```
traceroute localhost
```

atp..

Část I

Úvod

Virtualizace je již delší dobu známý pojem v oblasti výpočetní techniky, nebyla ale dříve tolik využívána kvůli vysoké režii samotných virtualizačních nástrojů. Ta doba je již naštěstí za námi a nyní je možno věnovat výpočetní výkon počítače především operačním systémům ve virtuálním počítači a ne virtualizaci samotné. Díky tomu se naskýtá mnoho možností, kde je možno virtualizaci využít.

Jednou z možností je virtualizace na desktopu, kdy má každý virtuální počítač své okno a mezi těmito okny (počítači) se může neomezeně přepínat, nastavovat a sledovat chování. Ve své bakalářské práci se zabývám skutečností, že je možné tyto virtuální počítače různě zapojovat do virtuálních či skutečných sítí. Na síti virtuálních počítačů je možné se naučit nastavování jednotlivých serverů či klientů a je jednodušší vše pochopit, když je možné mít vše najednou před očima na jediném počítači, než mít počítače rozmístěné po celé učebně či ještě hůře, po celé škole.

Výborným nástrojem na virtualizaci je KVM. Nabízí mnoho možností k nastavení, není náročný a nechá se s ním spustit skoro každý operační systém, proto si zaslouží, abych se mu ve své práci věnoval. Virtuální počítače se do sítě nespojí sami od sebe, musí to udělat uživatel. Několik takových možností popisuji ve svém souhrnu praktických tipů pro výuku sítí podpořenou virtualizací, doplňuji ukázkami kódu v BASHi, který pro přehlednost popisuji.

Vzhledem k tomu, že na většině síťových prvků v internetu je nějak upravený Linux, předvádím tvorbu sítí na Linuxu, to ovšem neznamená, že není možné virtualizační nástroj nahrát na jiný operační systém, či jiný operační systém do něj.

Virtualizace se nechá využít i pro tvorbu virtuální desktopové infrastruktury, kdy je možné přesunout výpočetní operace z klientských stanic na vzdálený server. Toto řešení se přímo hodí do škol. Virtuální desktopové infrastruktury se ve své práci věnuji na systémech od společností VMware ESXi a CITRIX XenServer, společně s komplexním řešením VDI-in-a-box.

pro čtení - tam se přepínač -snapshot použít musí. Scripty pro spouštění více počítačů s nastavením pro různé situace si předvedeme později.

Zkusme si teď spustit virtuální počítač, který jsme si nainstalovali.

```
kvm -hda virtualni_disk.img -m 256M
```

Příkaz jistě není potřeba vysvětlovat, vše již známe.

1.2.4 Propojení více virtuálních počítačů do virtuální sítě

Předvedu dvě možnosti, jak je možné připojit více virtuálních počítačů do virtuální sítě. První bude pomocí TCP socketu, druhá virtuální síť bude tvořena multicastem. Síť tvořená TCP socketem má větší latenci, zato vyšší přenosovou rychlost. Naopak virtuální síť tvořená multicastem má velmi nízkou latenci a nižší rychlost. Pokud budete chtít mít v síti virtuální počítače s nějakým OS od Microsoftu, volte první možnost. Multicastové propojení spojí virtuální počítače na stejném multicastu a portu na všech hostitelích v síti.

Pomocí TCP socketu První možností je připojení pomocí TCP socketu, kdy jeden virtuální počítač naslouchá na daném TCP portu a ostatní počítače se na tento port připojují přes loopback hostitelského PC.

První virtuální stroj spustíme příkazem (VM0)

```
kvm -hda linux.img -m 256M -snapshot \  
-net nic,macaddr=52:54:00:11:22:33,model=virtio \  
-net socket,listen=:1234
```

1 Cíle práce

1. Otestovat programy pro desktopovou virtualizaci (VMware workstation, Win Virtual PC, KVM).
2. Na základě testování vybrat jeden program, o kterém napíšu souhrn praktických tipů pro učitele sítí.
3. Otestovat řešení pro virtualizaci serverů a virtuální desktopovou infrastrukturu (platformy VMware a Citrix).
4. Na základě testování vybrat jednu platformu, o kterém napíšu souhrn praktických tipů na vytvoření virtualizované učebny.
5. Sepsat souhrn praktických tipů.

Přepínači `-hda` a `-cdrom` určíme soubory s virtuálním a instalačním diskem. `-boot d` říká, že se má systém zavádět z cdromu, není ovšem povinný, pokud jde o instalaci na prázdný virtuální disk (nový/bez zaváděcího záznamu).

Přepínačem `-m` zvolíme, s jak velkou pamětí RAM virtuální počítač spustíme. Hodnoty můžeme udávat v M nebo G, pokud neudáme nic, využije se 128MB paměti.

Po potvrzení příkazu se spustí virtuální počítač s naboootovaným instalačním diskem. V instalaci postupujeme stejně, jako bychom instalovali operační systém na fyzický počítač.

Co s nainstalovaným systémem? Hned po instalaci je vhodné si udělat zálohu virtuálního disku. Např takto:

```
cp virtualni_disk.img debian_6_po_instalaci.img
```

Na této záloze nebudeme provádět žádné změny, využijeme jí pouze v případě, když budeme chtít pracovat na čerstvě nainstalovaném systému, a to tak, že si ji překopírujeme. Tímto krokem se vyhneme opakované instalaci a ušetříme si čas.

Je mnoho možností, jak uchovávat a spouštět virtuální stroje na více počítačích - řekněme ve školní třídě. Nabízí se třeba možnost rozkopírování virtuálních disků do každého počítače, kde si každý student nastaví virtuální počítač(e) podle sebe a jejich nastavení bude uchovávat na svém disku, nebo můžeme image s virtuálními počítači nechat pouze na síťovém disku a spouštět je z něho. Určitě nebudeme chtít, aby se měnil virtuální systém na síťovém disku, proto k němu budeme přistupovat programem KVM jedině s přepínačem `-snapshot`, který zajistí, aby se všechny změny ukládaly jen jako dočasné soubory a nepřepisovaly soubor s virtuálním diskem. Pro jistotu je dobré mít síťový disk, ke kterému přistupuje více uživatelů s právy pouze

Část II

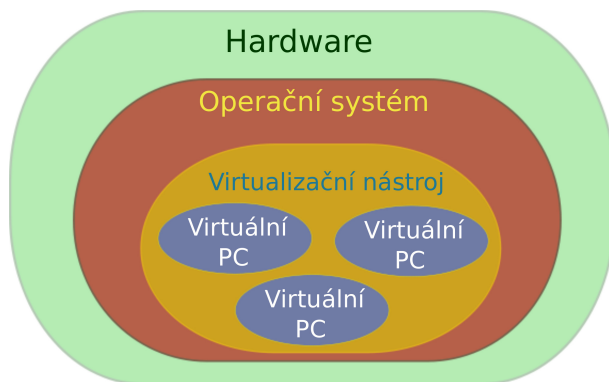
Teoretická část

V teoretické části se věnuji možnostem virtualizace na desktopech a na serverech. Popisuji typy virtualizace, jejich výhody a nevýhody, představuji software, který jednotlivé typy virtualizace reprezentuje.

2 Virtualizace na desktopech

Host-based virtualizace Když mluvím o virtualizaci na desktopech, myslím tím tak zvanou host-based virtualizaci. Jedná se o takovou virtualizaci, kdy na fyzickém počítači je nainstalován běžný operační systém a až v něm je spuštěn virtualizační nástroj - jako na obrázku 1. Toto řešení má obrovskou výhodu v tom, že na jediné obrazovce můžeme vidět najednou okna se všemi virtuálními počítači a správa veškerých strojů se může provádět po pouhém překliknutí do jiného okna. Jak to na světě běžně chodí, většina výhod souvisí s nějakými nevýhodami, v tomto případě tomu bohužel není jinak. K tomu, abychom měli takovéto pohodlí, potřebujeme již zmíněný operační systém s grafickým rozhraním a to spořádá část výkonu počítače. Není tedy na škodu věnovat pozornost snížení takovéto režie.

Převážně host-based virtualizaci se věnuji ve své práci.



Obrázek 1: Host-based virtualizace

Příkazem `create` říkáme programu `kvm-img`, že bude vytvářet nový image disku.

Parametrem `-f` nastavíme formát virtuálního disku. Tímto parametrem neformátujeme oddíly na disku, ale volíme formát image disku - jednoho souboru. Výběrem formátu můžeme ovlivnit, jak bude virtuální disk rychlý, nebo třeba kolik bude zabírat místa na disku fyzickém. Formát `qcow2` volím z důvodu, že mi virtuální disk bude zabírat na disku pouze tolik místa, kolik bude virtuální disk obsahovat dat. Pokud bych použil např. formát `raw`, zabíral by obraz disku tolik místa, jak velkou kapacitu by měl, ovšem byl by o něco rychlejší, jelikož by se systém nemusel zatěžovat (de)kompresí. To znamená, že virtuální disk o zvolené velikosti 20GB s nově nainstalovaným linuxovým systémem bude zabírat na fyzickém disku zhruba 800MB (dle toho, co vše instalujeme) při formátu `qcow2` a 20GB při formátu `raw`.

Načítání dat z obrazů formátu `qcow2` výrazně zrychlíme, když aktivujeme prealokaci `-o preallocation=metadata`, dále už jen zvolíme název a velikost virtuálního disku. Velikost můžeme udat v jednotkách b, K, M, G, T.

Instalace systému Instalace operačního systému probíhá skoro stejně, jako na fyzickém stroji. Nejprve je nutno získat instalační cd, nebo lépe iso obraz instalačního cd - instalace z iso souboru proběhne rychleji, než z cd. U většiny linuxových distribucí se nechá stáhnout obraz s aktuálním systémem z internetu. V případě, že si chceme udělat iso soubor z CD, můžeme to udělat z příkazové řádky např. programem `dcfldd` v podobě

```
dcfldd if=/dev/cdrom of=obraz_disku.iso
```

Když máme nachystané obrazy disků, můžeme začít instalovat virtuální počítač.

```
kvm -hda virtualni_disk.img -cdrom instalacni_cd.iso -boot d -m 1024M
```

2.1 Operační systém

Host-based virtualizaci je možné provozovat na různých operačních systémech vcelku bez problémů. Uvedeme si dvě nejdostupnější možnosti - Linux a Windows. Obě možnosti mají svá pro a proti, zde si o nich něco povíme.

2.1.1 Linux

Linux se instaluje v podobě různých distribucí, některé jsou vybaveny od základu více programy, jiné méně, vybrat je možno i grafické prostředí. Z toho je zřejmé, že je snadné minimalizovat potřeby samotného operačního systému tím, že využijeme nenáročnou distribuci s méně náročným grafickým prostředím.

Jako vhodná distribuce se nabízí Debian ve větvi stable či testing. Stable nabízí stabilní a dobře otestovaný software a je s ním zaručen chod systému (i) za cenu, že ne vždy obsahuje nejaktuálnější verze softwaru. Větev testing obsahuje aktuálnější verze softwaru, které ale nejsou v systému otestovány tak dokonale, jako ve verzi stable, proto se kvůli nim mohou vyskytnout komplikace a jako možná větší nevýhodu je možno brát, že tato větev není podporována bezpečnostním týmem.

Nechalo by se říci, že Debian ve verzi stable se nabízí na počítač, u kterého je důležité, aby byl stále funkční i dlouhodobě bez dohledu, na kterém nepotřebujeme nové funkce programů a stačí nám to, co umějí otestované verze, naproti tomu testing je vhodný, pokud si chceme hrát i s novějšími verzemi programů, což může být pro virtualizaci klíčové, protože je to odvětví, které se poslední dobou velmi rozvíjí.

Grafické prostředí je vhodné vybírat co nejméně náročné, přece jen jde pouze o virtualizaci a není třeba si užívat všelijaké grafické efekty, které nabízejí “větší” grafická prostředí. Pro tyto účely je vhodné prostředí LXDE

stažení i instalační příručku najdete například na stránkách www.debian.org. V textu tedy budu uvádět příkazy určené pro operační systém Debian, v ostatních distribucích se kód může mírně lišit.

Když máme nainstalovaný operační systém, je ještě potřeba doinstalovat kompletní virtualizační řešení KVM. To uděláme příkazem:

```
apt-get install kvm
```

A pro jistotu zkontrolujeme, zda máme v jádře zavedeny moduly `kvm.ko` a `kvm-intel.ko` nebo `kvm-amd.ko` (podle procesoru)

```
lsmod | grep kvm
```

Pokud nám terminál vypíše dva řádky, které budou začínat jmény výše uvedených modulů, máme systém připraven pro virtualizaci.

1.2.3 Instalace virtuálního operačního systému

Virtuální disk Abychom mohli nainstalovat virtuální operační systém, potřebujeme virtuální disk (tzv. image disku), na který se vše nainstaluje. Virtuální disk bude na fyzickém disku reprezentován jedním souborem. Tvorbu disku provedeme programem `kvm-img` spuštěným z příkazové řádky v následující podobě.

```
kvm-img create -f qcow2 -o preallocation=metadata virtualniDisk.img 20G
```

nebo XFCE. Jedná se o velmi “lehká” prostředí, určená původně pro starší a pomalejší počítače, takže nezaberou mnoho prostředků a většina jich zbyde pro samotnou virtualizaci.

V mnoha případech virtualizace funguje hostující operační systém jako router, proto i v tomto ohledu je vhodné virtualizovat na Linuxu - na systému, který má k síťovým prvkům tak blízko.

2.1.2 Windows

U operačního systému Windows není možnost volby grafického prostředí¹ ani neinstalování součástí systému, které nebudou využívány. V dnešní době, když je známo, kdy Microsoft ukončí podporu Windows XP, je asi jediná rozumná volba při výběru operačního systému od Microsoftu instalovat Windows 7.

V případě plánovaného užívání Windows virtual PC jsou vhodné verze Profesional, Enterprise či Ultimate, které tento produkt podporují.

I přes to, že nelze nároky operačního systému snížit tak, jako v případě Linuxu, je možné zakázat některé nepoužívané funkce a služby, případně upravit vzhled a tím ony nároky alespoň částečně snížit.

¹Existují nějaké pokusy se změnou grafického prostředí, ale není to nic rozumného, navíc se tím porušují licenční podmínky.

1.2 Návrh řešení na platformě Linux

1.2.1 Hardwarové vybavení počítače

Hardwarové parametry fyzického počítače volíme podle toho, kolik virtuálních počítačů budeme chtít na daném počítači spouštět a jaké operační systémy to budou. Zjednodušeně se dá říci, že operační systém s virtualizačními nástroji využije 10-15% výkonu fyzického počítače, o zbytek se mohou podělit virtuální stroje. Procesor je srdce počítače a v tomto případě to platí několikanásobně, protože budeme spouštět více virtuálních počítačů, je tedy vhodné vybírat výkonné procesory a vzhledem k adresaci více RAM volit 64b architekturu. Pro naše účely je důležité, abychom měli v počítači procesor s podporou hardwarové virtualizace. U procesorů Intel se technologie nazývá Intel Virtualization Technology (Intel VT), u procesorů AMD se technologie nazývá AMD-V. Minimální kapacitu paměti RAM a pevného disku zvolíme tak, že sečteme, kolik jí potřebuje samotný hypervizor a jednotlivé virtuální stroje (u kapacity pevného disku záleží na použitém formátu obrazů - je možné zvolit kapacitu nižší). Základní deska musí být osazena patičkami pro vybraný procesor a umožňovat připojení dostatečného množství paměti RAM. Pokud budeme chtít, aby na každém počítači bylo nainstalováno 15 obrazů disků s linuxovým operačním systémem a 3 obrazy se systémem Microsoft Windows, z nichž se vždy 7 obrazů zazálohují a spustí, budeme potřebovat zhruba 360 GB disk, 8 GB RAM, dostatečně rychlý procesor a odpovídající základní desku. Udané hodnoty jsou minimální a spíše orientační - vždy je vhodné uvážit jaké virtuální systémy se budou spouštět a počítač sestavovat tak, aby byl výkon vyšší.

1.2.2 Softwarové vybavení počítače

Distribuci operačního systému doporučuji volit podle zkušeností s ní a nároků na hardware. Pokud nemáte zkušenosti s žádnou distribucí, nebo se nemůžete rozhodnout, doporučuji Debian s grafickým rozhraním XFCE. Distribuci ke

2.2 Virtualizační nástroje

Zde uvádím několik nástrojů, které je možné využít pro host-based virtualizaci. Každý nástroj má svá specifika, která uvádím.

2.2.1 KVM

KVM v tomto případě nemá nic společného s KVM Switchem², je to velice kvalitní virtualizační nástroj pro Linux, ve kterém je možno virtualizovat nepřeberné množství operačních systémů. Zkratka KVM znamená Kernel-based Virtual Machine, tedy virtualizace založená na podpoře v jádře³. KVM je v linuxovém jádře od verze 2.6.20, tudíž dnes snad ve všech běžných linuxových distribucích.

KVM vychází z emulačního nástroje QEMU, je to v podstatě to samé, jen s tím rozdílem, že KVM virtualizuje, kdežto QEMU emuluje a KVM vyžaduje podporu virtualizace přímo v procesoru⁴, proto je také rychlejší. U procesorů Intel se technologie nazývá Intel Virtualization Technology (Intel VT) a u procesorů AMD se nazývá AMD-V.

KVM nabízí mnoho možností k nastavení. Není problém připojit k virtuálnímu počítači více disků, CDROMů, paměťových karet atp. Nastavení z jakého média se má počítač bootovat, připojit do virtuálního počítače virtuální procesor(y) s počtem jader dle přání, grafickou kartu nebo síťový adaptér dle vlastního výběru či připojení USB zařízení také není problém. KVM nabízí mnoho možností pro práci s okny, a grafickým výstupem. Například je možné vypnout grafický výstup virtuálního počítače úplně a přistupovat k němu přes VNC klienta.

Obrovskou zbraní tohoto virtualizačního nástroje je práce se sítí. Jak jsem již zmínil, je možné virtuální počítače spouštět s různými síťovými kartami, mezi které patří i karta virtio, což je virtuální karta vytvořená speciálně pro virtuální počítače. Ovladače pro tuto kartu jsou v linuxovém jádře obsaženy a do systému Windows je také není problém nahrát, jedná

²Zařízení pro připojení jedné klávesnice, monitoru a myši k více počítačům. KVM - Keyboard, Video, Mouse

³Volný překlad

⁴V procesoru jsou implementovány pomocné instrukce

1 Virtualizace při výuce sítí

1.1 Popis problému

Praktická výuka sítí ve škole předpokládá dobré a většinou i nákladné vybavení. To je zejména kabelové propojení počítačů, routerů, switchů a serverů, samotné počítače a servery, jiné síťové prvky a v neposlední řadě také prostory. Vzhledem k tomu, že pokud bude chtít více studentů nastavovat servery ve stejný čas, bude zapotřebí i více serverů, než jen jeden.

A co teprve časová náročnost na vrácení serveru (i jiných síťových prvků) do původní podoby, aby se další student nemusel trápit s tím, co předchozí student přenastavil? Pokud je to vůbec v našich silách, zajisté to bude zbytečně dlouho trvat a bude potřeba vynaložit nemalé úsilí pro objevení změn od původního nastavení.

Představa takovéto výbavy a časové náročnosti může nejednu školu od výuky sítí odradit. Při použití virtualizace nám většina těchto problémů odpadne a pro jednoho žáka bude postačovat pouze jeden výkonnější počítač.

Pro použití virtualizace ve výuce sítí připadá v úvahu virtualizační nástroj KVM na Linuxovém operačním systému.

se o takzvanou paravirtualizaci, síťová karta je pak rychlejší, než emulovaná skutečná karta. Virtuální počítače je možné mezi sebou propojit do virtuální sítě⁵ například multicastem či TCP Socketem a je to možné provést i s virtuálními počítači na jiném hostujícím⁶ počítači. Takto vytvořenou virtuální síť je nadále možné propojit do skutečné sítě a využívat všech možností, které síť nabízí. Pohyb packetů na síťových kartách je možné zaznamenávat do souboru na fyzickém disku a využít je k analizi sítě v některém z analizačních programů.

Virtuální systém je možné spouštět i bez zásahu do souboru s virtuálním diskem, na němž je systém nahrán. Díky tomu je možné se systémy laborovat a v případě, že se něco nepovede jen znovu spustit systém.

Spouštění virtuálních strojů se provádí z příkazové řádky, případně z nějakého grafického uživatelského rozhraní, které stejně pracuje nad těmito příkazy. Pro rychlou a pohodlnou práci s jedním virtuálním počítačem (nebo málo počítači) se nabízí využít ono GUI, ale pro spouštění a konfigurování více počítačů ve více sítích je nejlepší využít BASHi a napsat scripty ke spuštění.

2.2.2 VMware Workstation

Tento virtualizační nástroj je vyvíjen firmou VMware, zabývající se nejen host-based virtualizací, ale také nativní virtualizací.

VMware Workstation je profesionální, velmi propracovaný nástroj určený pro Linux a Windows. Ani zde není problém nainstalovat většinu aktuálních operačních systémů a přidat k virtuálnímu počítači pevné disky, CD-ROMy, diskety, nebo třeba USB zařízení.

Také je v tomto případě možné využít podporu virtualizace v procesoru a tím virtualizaci urychlit. Před instalací známého operačního systému⁷ program nabídne doplnění údajů jako například jména a hesla uživatelů, licenční klíč atp, není potom potřeba vyčkávat u počítače během instalace, protože VMware Workstation tyto údaje automaticky doplní.

⁵Klidně do několika sítí.

⁶Fyzickém

⁷Pro VMware Workstation známého systému. Seznam známých operačních systémů je skutečně veliký a bez problémů se tam najde většina běžně používaných OS.

Souhrn praktických tipů nejen k výuce sítí podpořené virtualizací

Lukáš Komárek

Zacházení se sítí je zde také na vysoké úrovni. Virtuální počítač je možno jedním kliknutím zapojit do sítě například přímo, skrýt za NAT, propojit pouze s hostitelským systémem nebo pouze s ostatními virtuálními systémy do takzvaného segmentu. Téměř všechnu práci se sítí dělá VMware Workstation automaticky.

Program se ovládá pomocí GUI, kde má každý virtuální počítač mimo tlačítka play na spuštění také svojí záložku s nastavitelnými parametry a náhledem systému.

2.2.3 Windows Virtual PC

Virtualizační nástroj určen pro Windows 7⁸ řady Profesional a vyšší, vyžaduje podporu virtualizace v procesoru. Je s ním možno virtualizovat především operační systémy od Microsoftu. Je určen pro 32b systémy(virtualizované). Virtuální počítač je možno připojit do interní sítě, skrýt za NAT, či mu připojit jednu ze síťových karet. K virtuálnímu počítači je možno připojit CD a DVD z iso souboru či přímo z mechaniky, stejně tak i obraz pevného disku.

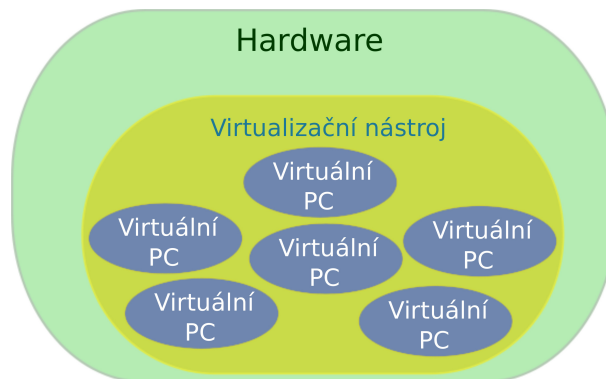
Tento program je zaměřen převážně na uživatele přecházející ze systému Windows XP na Windows 7, aby mohli používat software, na který jsou zvyklí.

3 Nativní virtualizace

Nativní virtualizace je taková, kdy virtualizační nástroj ovládá hardware přímo a ne přes operační systém, resp. operačním systémem je v tuto chvíli onen virtualizační nástroj. Výhoda oproti host-based virtualizaci je očividná - virtualizační nástroj má naprostou kontrolu nad hardwarem, tudíž rozděluje prostředky přesně tak, jak potřebuje, navíc na počítači neběží operační systém, který by využil část výkonu. Na obrázku 2 je vidět, že při této možnosti je více prostředků pro virtuální počítače. Architektury jednotlivých řešení se lehce odlišují. Tak, jak je to naznačené na obrázku

⁸Jsou i jiné verze programu určené pro starší OS Windows, mají ale lehce jiné jméno.

je to například u XenServeru. V některých systémech přistupují virtuální počítače k některým částem hardwaru přímo, obsahuje tedy každý virtuální počítač jiné ovladače. Tato situace se ovšem určitě netýká procesorů či operační paměti, ty ovládá hypervizor vždy.



Obrázek 2: Nativní virtualizace

3.1 Virtualizační nástroje

3.1.1 VMware ESXi

Tento nástroj je určen pro virtualizaci počítačů s operačními systémy, jejichž řádka je skutečně velká. Je určen pro servery s x86 architekturou a využívá technologie Intel VT nebo AMD-V. Ovládá se softwarem vSphere Center z operačního systému MS Windows XP a novějším. Jak se dá od společnosti VMware očekávat, je hypervizor s ESXi velmi propracovaný a při práci s virtuálními počítači není administrátor ničím limitován.

Virtuální počítače je možné zapojovat do sítě dle libosti - připojení k virtuálním switchům či routerům během několika kliknutí a tím přenést zátěž z fyzické sítě na server.

ESXi se instaluje na disky se souborovým systémem VMFS, který je navržen přímo pro virtualizaci. Celý systém může fungovat na discích SAN.

Pro účel této bakalářské práce je podstatné, že je možné na ESXi provozovat VDI-in-a-Box a díky tomu zajistit virtuální desktopovou infrastruk-

туру.

3.1.2 CITRIX XenServer

I tento systém je určen pro nativní virtualizaci na x86 architektuře, vyžaduje technologie Intel VT nebo AMD-V a je rovněž velice profesionální. Management hypervizoru se provádí programem CITRIX XenCenter nainstalovaným na OS MS Windows 2000 a mladší. Pokud bychom chtěli porovnávat seznam podporovaných operačních systémů pro virtualizování s VMware ESXi, byl by seznam XenServeru kratší, i přes to je podpora velice dobrá - UNIXové OS podporuje XenServer pouze experimentálně.

Dle očekávání, společnost CITRIX také připravila VDI-in-a-box pro svůj XenServer, tudíž ani s VDI není na XenServeru problém.

4 Jiné případy virtualizace

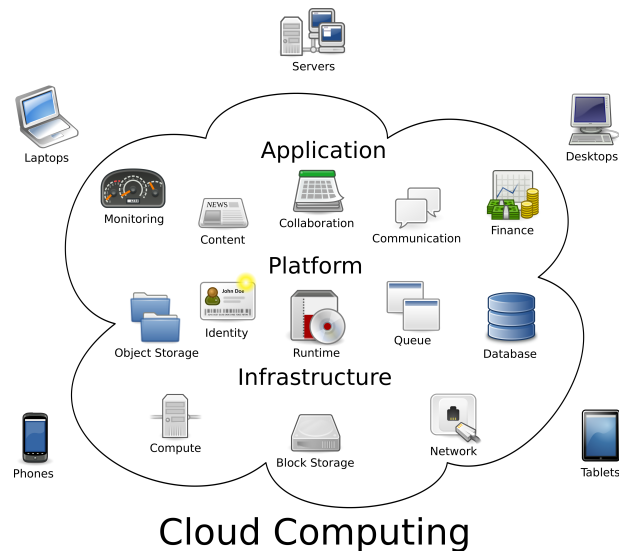
4.0.3 Cloud computing

Při využívání Cloud computingu se uživatel nemusí starat o hardwarovou stránku věci, může ho zajímat pouze software či prostředky, které využívá a dynamicky je měnit. Jde o to, že uživatel, který má v Cloudu například virtuální server, má najednou potřebu přidat do serveru procesor, více operační paměti nebo zvětšit disk, proto požádá poskytovatele Cloudu a ten během chvíle přiřadí danému serveru požadované prostředky. Obdobně to funguje i s aplikacemi - uživatel si sám určí, k jakým aplikacím na serveru bude moci přistupovat. O běh aplikací se stará správce Cloudu a uživatele může zajímat pouze funkčnost. Výhoda takového přístupu je ta, že uživatel Cloudu nemusí platit za software či prostředky, které nevyužívá a začne za ně platit až v momentě, kdy je skutečně potřebuje. Uživatel se nemusí se zajímat o instalaci, jde mu pouze o to, aby požadované věci získal snadno a rychle. Vzhledem k tomu, že aplikace „běží někde na internetu” se k ní může přistupovat odkudkoli, a třeba i z čehokoli (PC, tablet, telefon atp.). Pro lepší představu Cloud computingu může pomoci obrázek 3.

Nevýhody mohou plynout z nedostatečné kontroly nad verzemi nabízeného softwaru a celkově z toho, že se o chod systémů stará někdo jiný.

Cloud computing by bylo možné rozdělit podle toho, jaké služby jsou v Cloudu nabízeny:

- **SaaS** - Software as a Service - Na serveru běží aplikace, poskytovatel zajišťuje její chod, aktualizace atp, uživatel k ní přistupuje třeba přes webový prohlížeč.
- **PaaS** - Platform as a Service - V rámci Cloudu je poskytována celá platforma obsahující vše potřebné např. pro programátory.
- **IaaS** - Infrastructure as a Service - Poskytována je celá infrastruktura, například server s určitým operačním systémem.



Obrázek 3: Cloud computing, obrázek převzat z wikipedia.org

5 Výuka sítí podpořená virtualizací

Praktická výuka sítí ve škole předpokládá dobré a většinou i nákladné vybavení. To znamená zejména kabelové propojení počítačů, routerů, switchů a serverů, samotné počítače a servery, jiné síťové prvky a v neposlední řadě také prostory. Vzhledem k tomu, že pokud bude chtít více studentů nastavovat servery ve stejný čas, bude zapotřebí i více serverů, než jen jeden.

A co teprve časová náročnost na vrácení serveru (i jiných síťových prvků) do původní podoby, aby se další student nemusel trápit s tím, co předchozí student přenastavil? Pokud je to vůbec v našich silách, zajisté to bude zbytečně dlouho trvat a bude potřeba vynaložit nemalé úsilí pro objevení změn od původního nastavení.

Představa takovéto výbavy a časové náročnosti může nejednu školu od výuky sítí odradit. Při použití virtualizace nám většina těchto problémů odpadne a pro jednoho žáka bude postačovat pouze jeden výkonnější počítač.

Rozhodl jsem se z těchto důvodů sepsat souhrn praktických tipů pro virtualizaci podpořenou výukou sítí.

5.1 Výběr operačního systému

Nejdříve bylo nutné vybrat operační systém, na kterém budu virtualizaci předvádět, takzvaný hostující systém. Již výše jsem zmínil, že by bylo pro tuto příležitost vhodné mít méně náročný systém s jednodušším grafickým prostředím. Z tohoto důvodu vycházel lépe Linux.

Podstatné bylo, aby se přímo v systému nechala jednoduše, nejlépe s možností automatizace, nastavovat síť. K tomu je BASH jako dělaný, nechá se s ním nastavit NAT, routování či propojit síťová rozhraní do bridge v několika málo příkazech. Proto jsem se také přiklonil k Linuxu. Neříkám, že to není možné provádět na počítačích s operačním systémem Windows, jen si myslím, že je to na Linuxu srozumitelnější. Poslední slovo při rozhodování mezi Windowsem a Linuxem znělo jasně - Linux.

Při výběru distribuce jsem sáhl po distribuci Debian ve větvi stable. Rozhodl jsem se tak především z toho důvodu, že s Debianem mám největší zkušenosti a jsem zvyklý na jeho ovládání, neobsahuje mnoho zbytečného softwaru, který by pouze zpomaloval systém a stojí za ním obrovská komunita. Také vím, že veškerý potřebný software pro práci se sítěmi je buď v systému nainstalovaný, nebo je v oficiálních repozitářích. Zajisté by podobnou službu splnila i jiná distribuce, jen by v praxi bylo možná nutné doinstalovávat potřebný software, či psát do BASHe lehce odlišné příkazy - podstata věci by se ovšem neměnila.

Grafické prostředí jsem zvolil XFCE. Nebyl by určitě problém s jakýmkoli

jiným prostředím, toto jsem vybral z důvodu nízkých nároků. Jde v podstatě pouze o práci s okny, aby mohl být každý virtuální počítač ve svém okně a na jedné ploše takovýchto oken několik pro snadnou správu a kontrolu virtuálních počítačů. Nejde o žádné extra vychytávky či grafické doplňky v podobě ploch na 3D krychli, prezentaci oken v kartách nad sebou, průhledných oken ani všemožných stínů. Grafické prostředí instaluji hlavně kvůli pohodlí. Ve spoustě případů by bylo možné pracovat i bez grafického rozhraní a k počítačům přistupovat přes některý ze síťových protokolů jako například VNC či SSH, ale grafické prostředí ušetří dost času a studenti, učící se počítačové sítě na virtuálních počítačích, vše lépe pochopí, když uvidí na obrazovce okna, kde každé reprezentuje jeden počítač, než kdyby museli používat, jim doposud možná neznámé, protokoly či klávesové zkratky.

5.2 Výběr virtualizačního nástroje

Vzhledem k výběru operačního systému jsem přišel o možnost volby hypervizoru⁹ Windows Virtual PC. Nevybral bych tento program ani v případě, že by Linux jako hostující systém nebyl překážkou, a to zejména z důvodu, že je Windows Virtual PC orientovaný na virtualizaci operačních systémů převážně od Microsoftu.

Rozhodoval jsem se tedy mezi KVM a VMware Workstation, které se nechají všechny provozovat na Linuxu a je možno v nich spustit nepřeborné množství operačních systémů. Již jsem se zde zmiňoval o tom, že jsem se snažil, aby bylo vše možné ovládat přes BASH, tedy s jednoduchou možností scriptovat a tak nastavit třeba sesíťování virtuálních počítačů i jejich spuštění pomocí jednoho scriptu. To je hlavní důvod, proč jsem z výše zmíněných možností vybral KVM. Automatizace je u ostatních hypervizorů také možná, ale KVM + BASH se mi jeví jako nejlepší, když se s touto kombinací pracuje, nejlépe je při tom poznat, jak jsou virtuální stroje nastaveny a jak celek¹⁰ pracuje.

Nemyslím si, že cena softwaru by měla být vždy rozhodujícím faktorem, ovšem v tomto případě, kdy perfektně vyhovuje jedno zdarma dostupné řešení, je možná zbytečné platit za software, který pro naše potřeby udělá

⁹Virtualizačního nástroje.

¹⁰Virtuální počítače na jednom i více hostitelských počítačích, hypervizor, sítě atd.

stejnou práci. I to je důvod, proč jsem se nerozhodl pro VMware Workstation.

5.3 Pro koho je souhrn praktických tipů určen

Souhrn tipů je určen pro toho, kdo již zná základy sítě. Předpokládám, že jej bude využívat ten, kdo již umí, nebo je schopen se naučit, jak se například nastavují IP adresy na různých systémech, zejména na Linuxu, co je NAT a jak se zprovozní či mu nebudou cizí pojmy jako bridge, ping či multicast. Zkráceně řečeno, v souhrnu tipů nejen pro výuku sítí podpořenou virtualizací si kladu za cíl vysvětlit virtualizaci a ukázat jak je s její pomocí možno tvořit sítě, ne však učit základy sítí nebo operačních systémů. To vše snad není potřeba říkat, jelikož je souhrn tipů určen pro učitele sítí, který tyto vědomosti jistě má.

Ne pouze učitel sítí souhrn tipů využije. Dost informací v něm najde i ten, kdo si chce vytvořit síť virtuálních počítačů ať už pro studijní účely nebo pro ostré nasazení v síti.

5.4 Co je a není cílem sepsaného souhrnu tipů

Cílem souhrnu tipů je vysvětlit tvorbu sítí z virtuálních počítačů tak, aby byl čtenář schopen propojit virtuální počítače mezi sebou, nebo také s hostitelským počítačem. Čtenář bude obeznámen s možnostmi, které v ohledu práce se sítí virtualizace poskytuje a bude schopen je využít k tvorbě i rozsáhlejších sítí, třeba přes virtuální stroje na více hostitelských počítačích.

V žádném případě není cílem tvořit osnovu pro výuku sítí, nebo do ní jakkoli zasahovat. To také znamená, že není v mém souhrnu tipů psáno co učit a jak, ale je vhodná pro případ, kdy učitel chce vysvětlit látku, kterou by obvykle ukazoval na síti fyzických počítačů, ale rozhodl se využít virtualizaci. Nepředpokládám ani tvorbu sítí přesně podle návodu, ale spíše dávám prostor pro tvorbu řešení podle vlastních potřeb a ukazuji různé možnosti, které je možno kombinovat.

6 Vyrťualizovaná ućebna

Kdyř mluvíím o virtualizované ućebně, myslím tím virtuální desktopovou infrastrukturu. To je řpřipad, kdy na jediném výkonném serveru je spuřtěn virtualizační nástroj, který spouřťí virtuální desktopové systémy a uživatel s nevýkonným počítačem se k tomuto virtuálnímu PC může řpřipojit a pracovat s ním, jako s normálním výkonným počítačem. Je tedy možné touto cestou využít staré počítače, kterým již výkon na nové aplikace nestačí. Z tohoto důvodu si myslím, ře je toto řešení výhodné pro nejednu řkolu, proto vytvořím souhrn tipů pro virtualizovanou ućebnu.

6.1 Softwarové vybavení serveru

Virtuální desktopovou infrastrukturu zajistí kompletní řešení VDI-in-a-box od společnosti Citrix. Tento nástroj budu instalovat na hypervizory ESXi a XenServer. Tyto servery je dále možné využít pro virtualizaci dalších serverů, takže VDI nemusí být jediný důvod, proč virtualizaci na těchto platformách využívat.

6.2 Pro koho je souhrn tipů pro VDI určen

Nastavit VDI pomocí VDI-in-a-box není moc složité, dle mého návodu to zvládne člověk s průměrnými IT znalostmi.

Ćtenář by měl znát problematiku Active Directory, umět nainstalovat operační systém, měl by zvládat používat základní nástroje pro práci s PC, například používat editor registrů.

Virtuální desktopová infrastruktura je náročná na hardware, tudíž ten, kdo hodlá vytvořit ućebnu, by měl mít dostatečně výkoný server, který tyto operace zvládne.

6.3 Co je a není cílem sepsaného souhrnu tipů

Budu se snažit popsat základní nastavení serverové části i části s nástrojem VDI-in-a-box. Prostředí, ve kterém se bude spravovat hypervizor hodlám popsat tak, aby ětenář byl schopen tento nástroj využít nejen k instalaci VDI-in-a-box, ale také k virtualizaci jiných serverů.

Prostředí VDI-in-a-box hodlám popsat tak, aby čtenář věděl, co dělá a navedu ho k cíli, kdy do stávající domény Active Directory připojí právě onu virtuální desktopovou infrastrukturu.

Nekladu si za cíl popsat veškeré funkce použitého softwaru, spíše bych rád navedl čtenáře správným směrem s tím, že si případné doplňující informace bude schopen najít.

Část III

Praktická část

7 Cíle

Cílem práce je otestovat programy pro host-based virtualizaci VMware Workstation, KVM a Windows virtual PC. Na těchto nástrojích nainstalovat testovací operační systémy a prozkoumat možnosti, které daný software nabízí, zejména zjistit, jak je možné sestavit síť virtuálních počítačů. Vybrat jeden virtualizační nástroj a napsat souhrn tipů na tvorbu sítí mezi virtuálními počítači daným nástrojem tvořenými.

Další částí cílů je otestovat software pro nativní virtualizaci, VMware ESXi a CITRIX XenServer. Na tyto servery nainstalovat testovací systémy a vytvořit virtuální desktopovou infrastrukturu s OS Windows 7. Vybrat jeden virtualizační nástroj a sepsat návod na tvorbu virtualizované učebny jím realizované.

8 Testování virtualizačního softwaru pro desktopovou virtualizaci

8.1 Metodika

Test bude probíhat následovně: Nejdříve nainstaluji virtualizační nástroj do PC, zjistím, jaké možnosti nabízí, utvořím pomocí něho několik virtuálních počítačů a z nich vytvořím síť. Testovací síť bude jednoduchá, pouze 3 virtuální počítače v síti 192.168.10.0/24 a jeden z nich připojen k internetu jinou síťovou kartou. Počítač s připojením k internetu poskytne připojení ostatním. Jeden z virtuálních strojů s jedinou síťovou kartou bude se systémem Debian 6, druhý s Windows 7, router (Virtuální počítač se dvěma kartami) bude také se systémem Debian 6.

8.2 VMware Workstation

Hostující systém/PC

Debian 6 stable s kernelem 2.6.32.5 na PC s 64b procesorem Intel Core 2 duo obsahující technologie Intel VT - pomocné instrukce pro virtualizaci.

Získání programu

Program je dostupný na oficiálních stránkách VMware.com za cenu 199 USD nebo zdarma trial verze na 30 dnů. Stáhnutí vyžaduje registraci uživatele.

Instalace programu

Stažený soubor stačilo nastavit jako spustitelný, poté projít instalačním průvodcem a program se nainstaloval. Při prvním spuštění VMware žádal o hlavičky kernelu, které stačilo doinstalovat a bylo hotovo. Nainstaloval se program Workstation a vedle něj ještě Virtual Network Editor a Player.

Popis programu

Program ve verzi 8.0.2 nabízí hned na úvodní stránce tvorbu nového virtuálního počítače, otevření dříve vytvořeného počítače a připojení ke vzdálenému serveru. V levém sloupci je seznam již používaných/nainstalovaných virtuálních počítačů, které po rozkliknutí otevřou záložku s rychlým nastavením parametrů a náhledem systému. Tlačítkem play je možné virtuální počítač spustit a v záložce vidět na jeho obrazovku. Na spuštěném stroji je možné udělat snapshot, tedy rychlou kopii aktuálního stavu počítače, ke které je možné se později vrátit. Virtuální počítače je možné klonovat dvěma způsoby. Buď kompletním klonem, kdy se překopíruje celý virtuální počítač jinam, nebo pouze odkazem na klonovaný virtuální počítač se zaznamenáváním změn. VMware Workstation nabízí připojení ke vzdálenému serveru přes síť, je možné využít připojení k VMware Workstation, VMware ESX nebo VMware vCenter Server.

Instalace virtuálního počítače

S instalací nového počítače pomáhá uživateli jednoduchý průvodce. V

průvodci je možné nastavit, s jakými verzemi produktů od VMware bude tento počítač kompatibilní, přičemž je možné zjistit, jaké omezení bude tento počítač mít¹¹. Jako instalační médium je možné využít CD/DVD-ROM nebo iso obraz disku. Program buďto sám zjistí z instalačního média, jaký operační systém se bude instalovat, nebo je ho možné vybrat ze seznamu, který je skutečně rozsáhlý. Na základě tohoto výběru (rozpoznání) průvodce nabídne zadání údajů, které by jinak bylo nutné doplňovat až v průběhu instalace a zadá je při instalaci automaticky. Díky této možnosti se instalace urychlí, protože nebude instalátor muset čekat, než si uživatel všimne, že má zadat např. jméno a heslo. Dále je uživatel vyzván k tomu, aby řekl, kolik procesorů, jader a operační paměti bude ve virtuálním počítači chtít.

Při nastavování sítě při instalaci je možné si vybrat přemostěné zapojení, kdy se virtuální počítač připojí na fyzickou síťovou kartu a připojí se přímo do fyzické sítě (pokud na fyzické síti běží DHCP server, bez problémů od něj získá potřebná data), nebo je možné virtuální počítač pomocí NAT skrýt za IP adresu fyzického počítače a tak se připojovat do sítě, případně je možné vytvořit síť pouze mezi virtuálním(i) a fyzickým počítačem a jako poslední možnost je nepřipojovat virtuální počítač k síti vůbec.

Následně průvodce uživateli nabídne, jaký controller bude chtít v počítači mít pro SCSI disky, zdali chce vytvořit nový virtuální disk, použít již zhotovený nebo fyzický disk. Po nastavení disku program zobrazí shrnutí parametrů nového virtuálního počítače a nabídne spuštění stroje. Instalace probíhá normálním způsobem jako na fyzický počítač jen s tím rozdílem, že pokud uživatel zadal při tvorbě nového počítače informace pro pohodlnou instalaci, nebude tyto údaje požadovat.

Při instalaci virtuálního počítače s Windows 7 jsem si všiml, že je v systému upravené nastavení vzhledu, oproti jiným instalacím z téhož média. To napovídá tomu, že VMware Workstation upravuje alespoň některé systémy, aby nevyužívaly tolik prostředků a byly rychlejší.

Možnosti sítě

K virtuálnímu počítači je možné připojit více síťových karet a ty následně

¹¹Například maximální počet procesorů, velikost RAM, velikost disků atp.

připojovat do sítí, jak jsem již výše zmínil (přemostit, za NAT, pouze s počítači a ne ven), nebo je také možné nakonfigurovat síť pomocí nástroje Virtual Network Editor. Tento nástroj poskytuje dostatečný prostor k propojení počítačů do různých sítí dle vlastního přání. Je možné si vybrat, na jakou fyzickou síťovou kartu bude přemostění prováděno, zda má virtuální počítač získat IP adresy z DHCP serveru nebo jestli se má v hostujícím systému vytvořit virtuální síťové rozhraní v síti se síťovou kartou virtuálního počítače.

Tvorba sítě

V nástroji Virtual Network Editor jsem si nastavil jedno připojení vmnet1 jako host only - propojení virtuálních počítačů pouze mezi sebou a hostujícím počítačem. Nastavil jsem, aby IP adresy dával automaticky DHCP server v síti 192.168.10.0, vše jsem uložil a vytvořil klon počítače s Debianem. Dva virtuální počítače (win 7, Deb 6) jsem spustil s jednou síťovou kartou a přiřadil jí do sítě vmnet1. Třetí virtuální počítač (Deb 6, router) jsem spustil s dvěma síťovými kartami, přičemž jedna byla nastavena, aby se skryla za NAT a přistupovala k internetu, druhá byla také v síti vmnet1.

Virtuální počítače s jednou síťovou kartou dostaly IP adresu v síti 192.168.10.0/24, kdežto počítač, který má fungovat jako router dostal IP adresu pouze na síťovou kartu připojenou k internetu - 192.168.121.131. Druhé kartě jsem tedy nastavil IP adresu a zapl síťovou kartu:

```
ip a add 192.168.10.125/24 brd + dev eth1
ip link set up dev eth1
```

Poté jsem zkusil ping ze všech počítačů na všechny, který fungoval ve všech případech mimo pingů směřujících na virtuální počítač s Windows 7. Stačilo na tomto pc pouze vypnout firewall a spojení fungovalo.

Jal jsem se tedy zprovoznit na routeru překlad IP adres příkazy:

```
echo "1" > /proc/sys/net/ipv4/ip_forward
iptables -t nat -A POSTROUTING -o eth0 -j SNAT --to 192.168.121.131
```

Poté již stačilo nastavit výchozí brány a DNS servery na ostatních počítačích a internet byl zprovozněn.

Zhodnocení

VMware Workstation je mocný nástroj, jeho ovládání je jednoduché a intuitivní. Mnoho práce udělá automaticky a tím je schopen ušetřit spoustu práce. Troufnu si říci, že vytvoření virtuální infrastruktury na desktopu je s tímto nástrojem nejrychlejší ze všech testovaných.

Jako možná nevýhoda by se mohla projevit právě ona automatizace, kdy program upraví systém dle sebe a uživatel nemusí vědět, jak se to stalo. Pravděpodobně nevádí upravený vzhled¹², ale může padnout otázka: Kde to program ještě bez uživatelova vědomí dělá?

8.3 KVM

Hostující systém/PC

Debian 6 stable s kernelem 2.6.32.5 na PC s 64b procesorem Intel Core 2 duo obsahující technologie Intel VT - pomocné instrukce pro virtualizaci.

Získání a instalace programu

KVM je již obsaženo v linuxovém jádře, ale je potřeba doinstalovat některé součásti. V Debianu není potřeba přidávat žádné repozitáře, pouze spustit instalaci

```
apt-get install kvm
```

a vše se automaticky nainstaluje. V jiných distribucích to bude obdobné.

Popis programu

KVM je virtualizační řešení úplně jiného rázu, než předchozí program, ale i přesto je schopen plnit práci, co se virtualizace týče, výborně. KVM je postaveno na programu QEMU, rozdíl mezi těmito programy je ten, že QEMU počítače emuluje, kdežto KVM virtualizuje. KVM je obsažen již v linuxovém jádře a využívá pomocné instrukce procesoru. Ovládání probíhá pomocí příkazové řádky, je jednoduché a veškeré možnosti jsou skvěle popsané v manuálových stránkách. GUI je pro KVM dostupné též. Například

¹²Popáno v části *Instalace virtuálního počítače*.

virt-manager nebo AQemu a je s nimi možno nastavit mnoho parametrů, proto KVM nemusí odradit ani ty, kteří se bojí příkazové řádky.

Instalace virtuálního počítače

Před instalací je nutno připravit virtuální disk, na který se bude systém instalovat. KVM nabízí několik formátů disků. Domácí je qcow2, ale zvládá i formát Virtual Boxu od firmy Oracle - vdi, VMwareovský vmdk, vpc (VHD) pro řešení od společnosti Microsoft a jiné.

Disk se v příkazové řádce vytvoří například tímto příkazem:

```
kvm-img create -f qcow2 diskKVM.img
```

Instalace probíhá tak, že se KVM spustí normálním způsobem, jen s rozdílem, že se spustí s instalačním cd, takže příkaz vypadá zhruba takto¹³:

```
kvm -hda diskKVM.img -cdrom instalacniCD.iso
```

Dále pokračuje instalace stejně, jako na fyzický počítač. Po instalaci je na disku diskKVM.img celý systém, který je možno jednoduše spustit

```
kvm -hda disk.img
```

a pracovat v nově vytvořeném okně.

Možnosti sítě

Stejně jako u jiných nástrojů, i zde je možné spustit virtuální počítač s více síťovými kartami a jednotlivé karty zapojit vyloženě dle libosti. Jako příklad mohu uvést zapojení do TCP socketu, kdy se vytvoří virtuální síť mezi virtuálními počítači naslouchajícími stejnému portu. Dále je možné propojit počítače UDP multicastem vysílaným vybrané skupině virtuálních počítačů. Toto řešení se nechá využít k propojení virtuálních počítačů na více hostitelích. Případně je možné připojit síťovou kartu virtuálního počítače na virtuální síťové rozhraní v hostujícím počítači. Tyto věci se nechají kombinovat a vytvářet jimi skutečně rozmanité sítě.

¹³Vynechávám nepovinné parametry, které se doplní výchozími hodnotami, ale to neznamená, že nelze nastavit např. RAM, počet jader procesoru, vložit síťové karty atp.

Tvorba sítě

Síť se v tomto případě nevytvoří tak automaticky, jako ve VMware Workstation, problém to ale také není. Klonování/kopírování virtuálních disků není třeba provádět, stačí spustit virtuální počítač s parametrem -snapshot a veškeré změny se budou ukládat na fyzický disk pouze jako dočasné soubory.

Na hostujícím počítači se udělá překlad adres do sítě internetu příkazy

```
echo "1" > /proc/sys/net/ipv4/ip_forward
iptables -t nat -A POSTROUTING -o eth0 -j SNAT --to 192.168.0.1
```

a vytvoří se virtuální síťové rozhraní kterému se zadá IP adresa v jiné síti.

```
tunctl -t tap0
ip a add 10.0.0.1/24 brd + dev tap0
ip link set up dev tap0
```

K tomuto virtuálnímu rozhraní se připojí virtuální počítač, který bude dělat router mezi internetem a sítí virtuálních počítačů. Jak jsem již zmínil, bude mít jednu síťovou kartu připojen k rozhraní tap0, druhá bude připravena k navázání spojení pomocí TCP socketu. Každá síťová karta musí mít jinou mac adresu.

```
kvm -hda linux.img -snapshot \
-net nic,macaddr=52:54:00:12:34:56,vlan=0 \
-net tap,ifname=tap0,script=no,vlan=0 \
-net nic,macaddr=52:54:00:65:43:21,vlan=1 \
-net socket,listen=1111,vlan=1
```

V tomto pc je také potřeba nastavit IP adresu karty v internetu a „nahodit jí“, výchozí bránu a DNS. NAT se zde udělá stejně jako v řešení s VMware Workstation.

```
ip a add 10.0.0.2/24 brd + dev eth0
ip link set up dev eth0
ip route add default via 10.0.0.1
```

```
echo „nameserver 88.88.88.88” >> /etc/resolv.conf
echo ”1” > /proc/sys/net/ipv4/ip_forward
iptables -t nat -A POSTROUTING -o eth0 -j SNAT --to 10.0.0.2
```

Ještě se zprovozní síťová karta zapojená do virtuální sítě

```
ip a add 192.168.10.1/24 brd + dev eth1
ip link set up dev eth1
```

Zbylé počítače se spustí z hostitelského počítače těmito příkazy:

```
kvm -hda linux.img -snapshot \
-net nic,macaddr=52:54:00:11:22:33 \
-nat socket,connect=127.0.0.1:1111
```

```
kvm -hda windows.img -snapshot \
-net nic,macaddr=52:54:00:33:22:11 \
-nat socket,connect=127.0.0.1:1111
```

Nastaví se v nich IP adresy v síti 192.168.10.0/24, výchozí brána(192.168.10.1), DNS a jsou připojeni k internetu.

Zbytečně v tomto řešení dělám dvakrát překlad adres. Je to z důvodu, že se to snažím dělat podle toho, co jsem si před testem zadal, jinak by se to nechalo udělat jednodušeji. Toto řešení je v podstatě to samé, jako řešení sítě v předchozím případě (VMware), je to ale delší kvůli tomu, že to VMware dělá automaticky. Neboli - je potřeba připojit virtuální počítač k internetu s překladem adres, u VMwaru se klikne na položku NAT, zde se musí nastavit routování na hostovacím systému a výchozí brána a DNS na virtuálním stroji, který dělá router.

Zhodnocení

Možná se zdá virtualizace pomocí KVM zbytečně složitá, to zejména kvůli ovládání příkazy. Toho se ale není třeba obávat, naopak se toho nechá využít a vytvořit scripty na nastavení celé sítě a spuštění mnoha virtuálních počítačů. Při virtualizaci s KVM mám pocit, že vím, co se přesně děje a při případném problému jsem schopen zjistit příčinu a problém vyřešit.

8.4 Windows Virtual PC

Hostující systém PC s 64b procesorem Intel i3 duo obsahující technologie Intel VT - pomocné instrukce pro virtualizaci, Windows 7 Profesional

Získání programu Program je k dispozici na oficiálních stránkách Microsoftu.

Popis programu Program je jednoduchý, určen především pro instalaci virtuálních počítačů s OS Windows. Nabízí možnost připojení virtuálních harddisků, CD/DVD¹⁴, čtyř síťových karet.

Instalace systému Pokoušel jsem se instalovat virtuální počítač Debian 6 a stále se mi nedařilo, vždy to vyhodilo chybu. Potá jsem zjistil, že je nutné instalovat operační systém s 32 bitovou architekturou. Instalace poté proběhla bez problémů a systém se povedlo nainstalovat. Program nenabízí možnost klonování nainstalovaných systémů, tudíž je potřeba překopírovávat soubory s virtuálními disky skryté kdesi v systému.

Možnosti sítě Program umožňuje připojit 4 síťové karty a nabízí možnosti připojení na adaptéry v PC, sdílení internetu přes NAT, či interní síť mezi počítači.

Tvorba sítě Kvůli zaměření hlavně na Windows se mi nepovedlo připojit Linuxový systém Debian 6 připojit za NAT k internetu. Povedlo se mi připojit Debian k internetu pomocí přemostěného připojení, kdy ve virtuálním počítači byla připojena fyzická síťová karta. Tudíž jsem musel ustoupit

¹⁴i skutečných CD/DVD

a situaci částečně ulehčit. K takto připojenému počítači už jsem jednoduše připojil další virtuální počítače do jiné sítě a na virtuálním počítači jsem nastavil směrování stejně, jako v předchozích případech. Poté bylo možné připojit virtuální počítače pouhým zadáním výchozí brány a DNS.

Zhodnocení Toto řešení není vyložene špatné, nechalo by se říci, že je prostě zaměřené na jinou skupinu uživatelů. K výuce počítačových sítí by se program Windows Virtual PC nechal využít tak akorát na systémy od společnosti Microsoft.

8.5 Výběr nástroje

Pro tvorbu souhrnu praktických tipů jsem vybral virtualizační nástroj KVM. Upřednostnil jsem ho před ostatními programy z důvodu možnosti ovládání pomocí BASHe, možnosti instalování téměř jakéhokoli systému do virtuálního počítače a také kvůli ceně.

Windows Virtual PC nemá zdaleka tolik možností, co se výběru virtualizovaného systému týče, jako dva ostatní programy. Jeho zaměření je spíše na uživatele přecházející z Windows XP na Windows 7.

VMware Workstation je velice kvalitní nástroj, srovnatelný s KVM a jistě by ho bylo možné využít také, ale není tolik integrován do Linuxu jako KVM a jeho nevýhoda může být také cena 199 USD. Možnosti, které využijeme u VMware Workstation pro podporu výuky sítí získáme s KVM zdarma - tedy o 199 USD levněji.

9 Testování virtualizačních nástrojů pro nativní virtualizaci

9.1 Metodika

Test bude probíhat následovně: Nejprve nainstaluji virtualizační operační systém a nastavím potřebné údaje pro připojení hypervizoru k síti. Poté na osobní počítač nainstaluji program na správu serveru a pomocí tohoto softwaru v serveru vytvořím dva testovací virtuální stroje. V prvním bude nainstalovaný operační systém Debian 6, ve druhém MS Windows 7. Tyto počítače zapojím do stávající sítě, ve které je řadič domény Active Directory, virtuální počítač s OS Windows 7 zkusím do domény připojit, jako by byl připojen přímo do fyzické sítě. Mezi všemi počítači provedu testy spojení.

Do serveru nainstaluji CITRIX VDI-in-a-Box a pokusím se vytvořit virtuální desktopovou infrastrukturu s již vytvořeným virtuálním strojem s OS Windows 7.

9.2 VMware ESXi

Získání

VMware poskytuje ESXi zdarma, pouze s možností placené podpory. Instalační iso obraz se nechá stáhnout na stránkách vmware.com, kde je nutná registrace.

Pracoval jsem s verzí ESXi 5.0.0

Instalace serveru a zapojení do sítě

Server jsem nabootoval z disku s vypáleným iso obrazem, přivítala mne jednoduchá obrazovka s popisem verze, kde stačilo enterem potvrdit, že hodlám pokračovat v instalaci, dále jsem musel potvrdit EULA licenci. Mezi instalací na lokální disk nebo vzdálené úložiště jsem vybral instalaci na lokální disk, který ESXi využilo celý. Rozložení klávesnice je možné vybrat z obrovského množství, nechybělo ani české, proto jsem jej vybral a stiskl enter. Následovalo dvojí zadání hesla pro uživatele root a potvrzení varování, že disk bude rozdělen. To bylo vše, co chtěl instalátor ESXi vědět, dále již proběhla samotná instalace, po které stačil restart, aby bylo ESXi spuštěno.

Po spuštění serveru bylo důležité nastavit síť, aby se k serveru mohlo přistupovat vzdáleně. Proto jsem stiskl klávesu f2, zadal přístupové údaje (jméno: root, heslo takové, jaké jsem zadával při instalaci) a pomocí šipek jsem vybral možnost Configure Management Network a dále Network Adapters. ESXi nabízí možnost získání adres od DHCP serveru nebo nastavit pevně, já jsem se rozhodl, že zadám IP adresy pevně.

VMware vSphere client Server se ovládá vzdáleně programem vSphere Client, který pro svůj běh potřebuje Windows XP nebo novější. Z počítače s tímto systémem jsem se připojil pomocí prohlížeče na IP adresu serveru, odkud jsem instalátor vSphere Client stáhl. Instalace byla jednoduchá a po několikerém stisknutí tlačítka Next byl program nainstalován.

Po spuštění programu jsem vyplnil adresu serveru a přístupové údaje k administraci. Program se bez problémů připojil a mohl jsem jej využívat k administraci serveru.

Instalace virtuálního počítače a jeho zapojení do sítě

Příprava virtuálního počítače pro Windowsový i Linuxový systém probíhá v rámci ESXi stejně, pouze s rozdílem samotné instalace. Ta je ovšem zase stejná, jako instalace na fyzický počítač.

Z nabídky File jsem vybral možnost New Virtual Machine... a zvolil typickou instalaci. Jméno jsem napsal W7 pro první stroj, D6 pro druhý. Takto se nakonec zobrazují ve výpisu virtuálních strojů a i soubory těchto počítačů jsou v adresářích s těmito názvy v diskovém úložišti, které jsem vybral na další stránce. Podpora obou operačních systémů je obsažena v hypervizoru, proto jsem vybral na stránce Guest Operating System poprvé Windows 7 64, podruhé Debian 6 64. Do každého virtuálního počítače jsem vložil jednu síťovou kartu a zapojil do sítě VMnetwork0. ESXi nabízí 3 možnosti, jak VM zapisovat na disk, respektive jak bude reprezentován virtuální disk v souborovém systému datového úložiště. Možnost Thick provision Lazy Zeroed a Eager Zeroed zabere místa v datovém úložišti tolik, jak velký virtuální disk bude. Obě možnosti naalokují diskový prostor staticky, ale Lazy Zeroed nevpíše nuly do bloků, dokud se místo nebude skutečně využívat, kdežto Eager Zeroed vepíše nuly okamžitě. Možnost Thin Provi-

sion začne zabírat místo v datovém úložišti až tehdy, když na disk začneme zapisovat. Kvůli úspoře místa jsem zvolil možnost Thin Provision u obou VM. Ještě před dokončením jsem zaškrtnl možnost Edit the virtual machine settings before completion, abych mohl do virtuálních počítačů vložit instalační média. Ty jsem vložil do mechaniky serveru a v kolonce New CD/DVD jsem hodnotou Host Device řekl serveru, že má připojovat cd od sebe, zaškrtnutím Connect at power on jsem zajistil, aby se cd připojilo hned při startu VM. Tím nastavení virtuálních strojů končí, po spuštění VM jsem přes záložku Console nainstaloval systémy stejně, jako bych je instaloval na fyzický hardware. Zapojení virtuálních počítačů do sítě Server ESXi v defaultním nastavení připojuje virtuální počítače s virtuální kartou v síti VM-network0 rovnou do sítě, do které je připojen server, tudíž není zapotřebí jiného nastavení, když mi tato situace vyhovuje.

V síti je spuštěn DHCP server, a oba virtuální počítače od něj mají IP adresy. Ping mezi všemi počítači byl v pořádku a připojení počítače s W7 do active directory proběhlo naprosto stejně, jako u všech fyzických počítačů v síti.

9.3 CITRIX XenServer

Získání

CITRIX nabízí XenServer ke stažení zdarma, na stránkách CITRIXu není potřeba ani registrace.

Verze, se kterou jsem pracoval byla XenServer 6.1

Instalace

ISO obraz jsem vypálil na disk a nabootoval jsem server z něho. Instalátor hned začal zjišťovat konfiguraci serveru a poté jsem si vybral české rozložení klávesnice a potvrdil licenci. Poté jsem vybral instalaci na lokální disk a povolil Thin Provisioning, abych ušetřil místo v datovém úložišti a nechal jsem zkontrolovat instalační disk, abych se ujistil, že je vše v pořádku - kontrola nenašla žádnou chybu. Po vložení hesla jsem byl vyzván k zadání IP adres a nastavení časového pásma a NTP serveru. Po potvrzení se spustila instalace serveru. NTP server jsem nevyužil, proto jsem byl po instalaci vyzván k zadání správného času. K hotové instalaci stačilo již jen restartovat

server.

XenCenter

Program pro vzdálený přístup k XenServeru a jeho ovládání v GUI se nazývá XenCenter. Pro svojí instalaci vyžaduje operační systém Windows 2000 a mladší. Do počítače, ze kterého jsem chtěl XenServer spravovat jsem vložil instalační disk XenServeru a spustil instalátor XenCenteru. Samotná instalace byla rychlá a ničím záludná.

Po spuštění programu jsem pomocí možnosti Add New Server zadal IP adresu XenServeru a přístupové údaje k němu (známé z instalace). XenCenter se připojil k XenServeru a veškerou správu jsem již mohl provádět na dálku.

Instalace virtuálního počítače a jeho zapojení do sítě Tvorbu virtuálních počítačů jsem začal stisknutím tlačítka New VM. V nabídce operačních systémů byl jak Windows 7, tak Debian 6. Počítačům jsem zadal jména dle operačních systémů, které jsou v nich nainstalovány. W7 a D6. Stejně jako při instalaci virtuálních strojů na ESXi server, nastavení obou virtuálních počítačů proběhlo úplně stejně, proto není potřeba popisovat instalaci každého počítače zvlášť.

Na další straně jsem vybíral, kde se nachází instalační médium - na výběr jsem měl instalaci z CD/DVD mechaniky serveru, ze sítě a z lokálního disku. Zvolil jsem instalační médium vložené do mechaniky serveru a pokračoval nastavením operační paměti a procesoru pro počítače. Jelikož jsem při instalaci XenServeru povolil funkci Thin Provisioning, zapl jsem tuto možnost i při přípravě pevného disku pro tyto virtuální počítače.

Každému počítači jsem přiřadil jednu síťovou kartu a nechal jsem jí automaticky vygenerovat MAC adresu. Na další stránce jsem tlačítkem Create now vytvořil virtuální počítač, který začal hned bootovat z instalačního disku vloženého do mechaniky serveru a já jsem provedl samotnou instalaci v záložce console. Instalaci jednotlivých operačních systémů již není třeba popisovat, protože vše bylo stejné jako při instalaci na fyzický stroj.

Zapojení počítačů do sítě bylo i v tomto případě velice jednoduché - nainstalovaný počítač dostal přidělenou IP adresu od dhcp serveru, který běžel

venku před fyzickým strojem, tudíž se nemuselo provádět žádné nastavení počítačů ani serveru, aby se PC připojili do stávající sítě. Proto stejně jako v situaci, kdy jsem virtuální počítače měl na ESXi serveru bylo zprovoznění síťové části naprosto standardní.

9.4 VDI-in-a-Box

O VDI-in-a-box nepíší u každého serveru zvlášť z důvodu, že se instalace liší jen v drobnostech. Tyto odlišnosti popíší v této kapitole.

Získání

VDI-in-a-box jsem stáhl ze stránek citrixu jako zkušební verzi.
instalovaná verze byla 5.1

Instalace a nastavení

Do každého serveru bylo zapotřebí nainportovat předpřipravený virtuální počítač VDI-in-a-box. Zde je právě jedna odlišnost - do ESXi se importuje obraz ve formátu OVA, do XenServeru ve formátu XVA.

Import VDI-in-a-Box ve formátu OVA do ESXi proběhl rychle pomocí položky Deploy OVF Template... z nabídky file ve vSphere Center.

Do XenServeru to bylo možností import v nabídce File v programu pro ovládání XenServer - XenCenter.

Pro vytvoření virtuální desktopové infrastruktury jsem musel k VDI-in-a-Box přistoupit z počítače, na kterém byl nainstalovaný CITRIX Receiver - pro prvotní nastavení nebyl Receiver potřeba, ale pro vzdálené připojení k virtualizovanému desktopu při nastavování šablony ano. Citrix Receiver je dostupný na stránkách Citrixu a umožní přistupovat ke vzdálenému(virtualizovanému) desktopu přes internetový prohlížeč.

Samotný server VDI-in-a-box se nastavuje přes internetový prohlížeč také, ale dokud je práce jen ve webovém rozhraní, není Citrix Receiver potřeba. K webovému rozhraní jsem přistoupil zadáním adresy `https://<ip_adresa_virtualního_p`
`in-a-box>\admin` do prohlížeče firefox. Byl jsem vyzván k zadání jména a hesla, což je v defaultním nastavení admin (jméno), kaviza (heslo). Hned po přihlášení jsem byl vyzván k tvorbě nové virtuální infrastruktury. Po potvrzení, že chci vytvořit novou VDI jsem zadal informace o serveru (ESXi

nebo XenServeru), tedy jeho IP adresu a přístupové údaje, vybral jsem umístění diskového úložiště, na kterém budou obrazy virtuální desktopové infrastruktury. Dále jsem vybral vytvoření nového gridu a zvolil, že chci počítače připojit do domény Active Directory. toto nastavení ulehčí přihlašování uživatelů - není potřeba vytvářet nové uživatele a lze povolit přístup již vytvořeným uživatelům v AD. Na další stránce jsem hned zadal údaje o AD, tedy adresu, název a přístupové údaje. Průvodce nastavením mne donutil potvrdit, že ve své síti mám rezervované IP adresy pro virtualizované desktopy.

Dále jsem na serveru připravil nový virtuální počítač s os Windows 7 a do něj nainstaloval potřebné rozšíření. Zde je opět malá odlišnost mezi servery - do PC na ESXi serveru jsem nainstaloval VMware Tools a do PC na XenServeru jsem nainstaloval XenServer Tools. V PC na obou serverech se tyto nástroje instalují obdobně, tedy připojením virtuálního disku s instalátorem softwaru. Pak již stačilo software ve virtuálním počítači doinstalovat klasickým způsobem. Průvodce instalací doporučuje povolit ve firewallu porty 139, 1494, 2598, 3389 a 8080, já jsem vypl rovnou celý firewall v operačním systému Windows 7, protože mi tuto funkci splňuje firewall v síti. Dále jsem v nastavení virtuálního PC povolil vzdálenou plochu a přidal oprávnění pro uživatele ve skupině VDI_UZIVATELE. V řadiči domény jsem tuto skupinu přidal uživatelům, kterým jsem chtěl povolit přihlášení k VDI. Nastavení virtuálního počítače pokračovalo úpravou registru. Vytvořil jsem novou hodnotu

```
HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\
policies\system\LocalAccountTokenFilterPolicy
```

s údajem 1.

Pokračoval jsem v nastavení VDI-in-a-box zadáním jména pro tento obraz. Na další straně vyhodnotil systém můj obraz jako neimportovatelný, pomohl restart virtuálního počítače, zadal jsem informace k lokálnímu administrátorskému účtu a nechal VDI-in-a-box, aby si naimportoval nový stroj.

Při vytváření VDI na XenServeru mi operace skončila chybou instalace

desktop agenta, i v tomto případě pomohlo, abych virtuální počítač restartoval a opakovatel instalaci. Po úspěšném importu jsem se připojil k virtuálnímu pc již přes prohlížeč a zkontroloval, zda PC neobsahuje zařízení SVGA 3D (Microsoft Corporation - WDDM a v nastavení serveru jsem připojil k PC řadič USB. Vzhledem k tomu, že jsem chtěl, aby každý uživatel mohl mít své nastavení systému a svá data, zvolil jsem, že chci možnost pooled and personal desktops a potvrdil tlačítkem Prepare.

Když příprava obrazu skončila, přihlásil jsem se přes prohlížeč do virtuálního desktopu jako doménový uživatel, abych otestoval, zda je možné se k PC připojit a zda se nevyskytuje nějaký problém. Nevyskytoval.

Vytvořil jsem novou šablonu, které jsem zvolil název Žáci, jako prefix jsem zvolil zaci, suffix 000, dle toho vypadaly potom názvy virtuálních počítačů generovaných VDI-in-a-box. Povolil jsem připojování USB zařízení, zvolil maximálně 3 desktopy spuštěné z této šablony a vybral personalizaci desktopů. V řadiči domény jsem vytvořil skupinu zaci a vložil jsem do ní uživatele, kterým jsem chtěl povolit přístup k této šabloně a na záložce Users ve VDI-in-a-box jsem tuto skupinu také napsal. Když se mi na záložce Desktops zobrazilo, že již je připravený desktop k přihlášení, přihlásil jsem se přes webový prohlížeč (adresa serveru VDI-in-a-box) jako doménový uživatel a na obrazovce se mi spustil funkční desktop.

Přihlášení k VDI jsem zkusil i z tabletu s OS Android, do kterého jsem z obchodu nainstaloval Citrix Receiver a do tohoto programu zadal adresu VDI-in-a-box a rovněž uživatelské jméno a heslo doménového uživatele. OS Windows 7 byl v tuto chvíli i na tabletu.

9.5 Výběr hypervizoru

Rozhodl jsem se, že do souhrnu praktických tipů napíšu o VMware ESXi serveru.

Myslím si, že server se systémem XenServer nebude za ESXi zaostávat a zajisté hravě zvládnou oba nasazení na velké projekty. Při svém testování jsem nenarazil na žádnou jednoznačnou věc, která by bránila vytvořit VDI na řešení od té či oné firmy. Vybrat jsem tedy musel dle toho, jak se mi zacházelo s nástroji pro vzdálenou správu serveru a v této oblasti měl VMware vSphere Center výhodu jednoduššího ovládání.

10 Tvorba souhrnu praktických tipů nejen pro výuku sítí podpořenou virtualizací

10.1 Rozbor kapitol souhrnu praktických tipů pro výuku sítí podpořenou virtualizací

V této sekci popisuji, proč jsem se rozhodl vložit do svého souhrnu praktických tipů konkrétní kapitoly a jejich části. Některá tvrzení obsažená v textu odůvodňuji provedenými testy.

10.1.1 Popis problému

V kapitole popis problému se snažím vysvětlit důvod virtualizace při výuce počítačových sítí. Opodstatňuji to v první řadě tím, že není nutné pořizovat nákladné vybavení, které by se skládalo z potřebných síťových prvků pro jedno testování vynásobené počtem studentů.

Dále zmiňuji časovou náročnost na vrácení síťových prvků do původní podoby. To by bylo možné vyřešit z části i na nevirtualizovaných počítačích, kdy by se do fyzického stroje nahrál systém v podobě nějakého obrazu systému, nebo by se mohl připravit nějaký script vracející původní hodnoty. V případě scriptu by ale bylo nutné, aby žáci upravovali pouze určené hodnoty a nic jiného.

10.1.2 Hardwarové vybavení počítače

Zde pojednávám o parametrech, které by měl splňovat počítač určený pro výuku sítí podpořenou virtualizací. Nelze jednoznačně říci větu typu „Pořiďte si tento počítač, bude vám na virtualizaci stačit“, protože každý bude virtualizovat jiné systémy a jiný počet, snažím se tedy vysvětlit, jakým způsobem je možno se dopracovat k výběru správné konfigurace.

V kapitole užívám tvrzení, že operační systém s virtualizačními nástroji využije 10-15% výkonu počítače. To nelze říci nikdy přesně, už jen kvůli tomu, že na méně výkonných strojích bude procento vyšší a na hodně výkon-

ných zase nižší, ale pro představu to takto stačí. Proto také uvádím, že je to řečeno zjednodušeně. K číslu 10-15% jsem dospěl z měření výkonu v odborných člancích uváděných ve zdrojích.

Při pojednávání o procesoru zmiňuji doporučení 64b architektury z důvodu využití více operační paměti. Z důvodu, že se v souhrnu tipů nechci zabývat instalováním nestandardních součástí systému a rád bych se vyhnul zpomalování systému, vynechávám informace o tom, že by bylo možné použít jádro PAE, kde by toto omezení nebylo. Konkrétně v Debianu by to nic nestandardního nebylo, ale nesnažím se tento systém čtenářům nutit. Další důležitá věc, co se procesoru týče, jsou technologie Intel VT nebo AMD-V, které vysvětluji.

Dále uvádím příklad, jaké parametry by bylo vhodné zvolit v případě potřeby spouštět a uchovávat určité množství virtuálních počítačů, uvedené hodnoty jsou spočteny jednoduchou matematikou a doporučuji pro jistotu volit vyšší parametry, protože je vždy vhodnější sestavu předimenzovat, nežli poddimenzovat.

10.1.3 Softwarové vybavení počítače

Při výběru linuxové distribuce doporučuji v textu, ať čtenář sáhne po distribuci, se kterou má zkušenosti, i přes to, že já vše provádím na Debianu. To z důvodu, že se domnívám, že bude efektivněji pracovat ve známém prostředí, než kdyby instaloval novou distribuci a zvykal si na jiné ovládání. Věřím, že práci na jiné distribuci nebude ztěžovat ukazování příkazů pro Debian, protože uživatel, který ví co dělá, dokáže lehce pochopit, že konkrétní příkaz je ve formě pro Debian a má udělat nějakou věc, kterou je schopen napsat i na jím známé distribuci. Tyto domněnky jsou podloženy mými zkušenostmi z případů, kdy na internetu vyhledám nějaký návod určený pro jinou distribuci, než používám a jsem schopen bez problémů přepsat příkazy tak, aby jim rozuměla má distribuce. Čtenář, který nemá zkušenosti s žádnou distribucí by nejspíš nevěděl, jak příkazy psát pro jakoukoli distribuci, proto v tomto případě doporučuji používat stejný systém, jako používám já.

Instalací systému se nezabývám, jelikož to není mým cílem a předpok-

ládám, že učitel počítačových sítí, kterému je souhrn praktických tipů především určen, toto dokáže. Pouze v příkazech uvedu, že je nutno virtualizační nástroj nainstalovat a zkontrolovat jaderné moduly.

10.1.4 Instalace virtuálního operačního systému

Kapitolu o instalaci virtuálního počítače mám rozdělenou na tři části, to zejména z důvodu přehlednosti, protože je tato kapitola rozsáhlejší a čtenář by se v ní nemusel orientovat. Části jsou: Virtuální disk, Instalace systému a Co s nainstalovaným systémem?

Virtuální disk

Tuto část sepisuji z důvodu, že je nutné nejdříve vytvořit virtuální disk, na který se bude operační systém instalovat. Pro představu uvádím čtenáři, jak je virtuální disk na fyzickém disku reprezentován.

Nejdříve předvedu příkaz, kterým se virtuální disk vytvoří a poté popisuji jednotlivé volby. K vytvoření disku jsem použil program KVM-iso. Virtuální disky je možno tvořit i jinými programy, ale formát qcow2 umí pouze kvm-img¹⁵. Hlavně není důvod instalovat jiné programy, když KVM-img je již nainstalován a je určen právě pro KVM.

Popisuji všechny použité parametry v příkazu, aby čtenář věděl, k čemu co slouží a v případě, kdy by chtěl disk vytvořit s jinými parametry, věděl jak se to dělá. Ukázkový disk vytvářím ve formátu qcow2, pro to mám důvod ten, že jej doporučuje firma Red Hat, stojící za KVM, z důvodu zachování vysoké rychlosti i při malém objemu dat. Toto tvrzení je na oficiálních stránkách KVM, provedl jsem ale i ověření z jiného zdroje. Jedná se o test výkonu KVM s různými parametry, který uvádím mezi zdroji. Vlastní měření jsem neprováděl, toto ověření se mi zdá dostatečné. Výhody tohoto formátu ve svém souhrnu tipů uvádím též a porovnávám jej s formátem RAW, čtenář se tak bude moci rozhodnout, zda chce použít ten či onen. Ve výše zmíněných testech je informace o tom, že disk formátu qcow2 pracuje rychleji s volbou -o preallocation=metadata, proto tuto volbu uvádím.

¹⁵resp. qemu-img

Instalace systému

Čtenáři se v této kapitole pokouším vysvětlit, co vše je důležité k nainstalování virtuálního operačního systému. Říkám, že je potřeba sehnat si instalační médium pro požadovaný operační systém. Médium, které doporučuji je soubor iso, tedy obraz disku. K tomu mám důvod ten, že je instalace z obrazu rychlejší a pohodlnější, než z CD/DVD. Většina instalátorů operačních systémů se nechá z internetu stáhnout ve formátu iso, pokud ale někdo má systém pouze na cd a chce si vytvořit iso, uvádím příkaz, jakým to může provést.

Co s nainstalovaným systémem?

Poslední část této kapitoly jsem napsal z důvodu, aby měl čtenář jasnou představu, jak zacházet s nainstalovaným systémem. Rozhodl jsem se čtenáři doporučit, aby si udělal zálohu hned po instalaci systému. Tímto jednoduchým krokem, pouhým překopírováním virtuálního disku, si možná ušetří spoustu času. Vycházím z vlastních zkušeností, kdy jsem zálohu neprovedl, hodně jsem přenastavil systém a pak jsem chtěl spustit systém v původním stavu - musel jsem instalovat ten samý systém znovu, přitom stačilo udělat po instalaci klon systému.

Aby se čtenář nemusel trápit s překopírováváním souborů s virtuálními disky na na více počítačů, navrhuji v textu umístění souboru na síťový disk a spouštění virtuálních počítačů přímo z něho. S touto situací úzce souvisí potřeba použití parametru -snapshot, tudíž toho využívám a vysvětluji jeho použití.

V souhrnu tipů tvrdím, že při spouštění virtuálního počítače ze síťového disku, který je nastaven na read only, se parametr -snapshot použít musí. Toto tvrzení jsem si ověřil testem, který zde popíšu:

Síťový disk NFS, který je tvořen na počítači s operačním systémem Debian 6, je připojen přes síť lan k počítači se stejným systémem, který má být hostujícím pro virtuální počítač. Na síťový disk jsem nahrál soubor s virtuálním diskem, na kterém je nahrán také operační systém Debian 6 a síťový disk jsem nastavil pouze pro čtení. V hostujícím počítači jsem nejdříve spustil virtuální počítač s parametrem -snapshot a počítač se choval naprosto normálně, jak jsem předpokládal. Virtuální počítač spuštěn s tímto

parametrem ukládal data jako dočasné soubory v hostitelském počítači, tudíž nebyl žádný problém. Když jsem se pokusil spustit virtuální počítač bez onoho parametru, normálně naběhl pouze do menu GRUBu, dál už se spouštěl s výpisem chyb a nedostal jsem se ani k přihlášení uživatele. Ani toto chování mne nepřekvapilo - virtuální počítač bez parametru -snapshot se veškeré změny pokoušel ukládat na virtuální disk, ne pouze jako dočasné soubory do hostitelského pc, ovšem virtuální disk byl na disku určeném pouze pro čtení, tak tyto pokusy o zápis skončily chybami.

Na konec kapitoly uvádím příkaz na spuštění virtuálního počítače. Vyplněné hodnoty již nechávám bez popisu, protože jsem vše vysvětloval o pár řádků výše.

Propojení více virtuálních počítačů do virtuální sítě

V této kapitole konečně narážím na tvorbu sítí, obsahuje 3 části. První tvorba sítě Pomocí TCP socketu, druhá Pomocí UDP multicastu a nakonec Funkční síť. Hned v úvodu používám tvrzení, která jsem si ověřil testy. Jedná se o tvrzení:

- *Síť tvořená TCP socketem Má větší latenci, zato vyšší přenosovou rychlost.*
- *Síť tvořená UDP multicastem má velmi nízkou latenci, ale horší přenosovou rychlost.*

Důvody těchto tvrzení se nechají vyčíst z tabulky 2 a 3, kterou jsem sestrojil z naměřených hodnot.

Metodika testu

Test jsem provedl tak, že jsem propojil dva počítače s operačním systémem Debian 6 a 256MB RAM do virtuální sítě tvořené nejdříve TCP socketem, později UDP multicastem. Na prvním stroji jsem udělal síťový disk a na ten nakopíroval testovací soubor o velikosti 1,3GB, z druhého virtuálního

počítače jsem se připojil k síťovému disku a začal soubor kopírovat do domovského adresáře. Proces kopírování jsem měřil programem time a výslednou hodnotu zapsal. Pak jsem testoval odezvu mezi počítači programem ping a do tabulky napsal průměrnou hodnotu z dvaceti odeslaných/přijatých paketů. Testy jsem prováděl s virtuální síťovou kartou virtio a rtl8139 a vše provedl dvakrát. Počítače byli spouštěny s parametrem -snapshot a mezi každým testem jsem je vypnul a znovu spustil, tedy vrátil do původního stavu. Více už prozradí tabulky.

	TCP socket, virtio	TCP socket, rtl8139
Doba kopírování 1	1min, 40sec	2min, 11sec
Doba kopírování 2	1min, 44sec	2min, 18sec
Odezva 1 [ms]	43,729	41,569
Odezva 2 [ms]	40,913	38,51

Tabulka 1: Test TCP socketu

	UDP multicast, virtio	UDP multicast, rtl8139
Doba kopírování 1	2min, 47sec	3min, 56sec
Doba kopírování 2	2min, 49sec	4min, 1sec
Odezva 1 [ms]	1,179	0,529
Odezva 2 [ms]	1,129	0,656

Tabulka 2: Test UDP multicastu

Z tabulek je tedy možno vyčíst důvody psaných tvrzení a také důvod, proč je vhodnější používat paravirtualizovanou kartu virtio.

Dále v kapitole tvrdím, že když čtenář hodlá spojovat do sítě operační systémy od Microsoftu, měl by užít možnost s TCP socketem. Toto říkám z důvodu, že se mi nepodařilo propojit Windows XP ani Windows 7 do sítě tvořené UDP multicastem.

Pomocí TCP socketu

V této části nejprve v krátkosti popíšu, jak zapojení do TCP socketu funguje a poté již ukážu příkaz, kterým se první i další virtuální počítač spustí. Virtuální počítače pro přehlednost označuji jako VM0 a VM1. U příkazů na spuštění počítače záměrně vynechávám různé parametry, které

by dělaly příkaz košatější a tím pro čtenáře nepřehlednější. Snažím se, aby se příkaz, který píše, týkal pouze nastavování sítě, abych čtenáře zbytečně nevystavoval rušivým prvkům.

Příkazy na spuštění následuje popis jednotlivých voleb, mezi kterými je i nastavování síťové karty virtio. Nabádám čtenáře, aby tuto kartu do virtuálních počítačů zapojovali, protože je rychlejší. K tomuto tvrzení jsem došel při měření, které jsem zaznamenal do tabulek o půl stránky zpět.

Pomocí UDP multicastu

Tato část je vcelku krátká, protože jsem skoro vše potřebné vysvětlil v části o zapojení do TCP socketu. Zde už tedy pojednávám pouze o věcech týkajících se vyložení multicastového propojení virtuálních počítačů.

Funkční síť

Tuto část jsem do textu zařadil proto, abych předvedl, že ať už čtenář tvořil síť zapojením do TCP socketu, nebo multicastem, že je vše spojeno a nebyl to ani velký problém. Proto tedy jen v krátkosti nastavuji IP adresy a zkouším ping z jednoho počítače na druhý. Jak jsem již výše říkal, používám označení virtuálních počítačů VM0 a VM1 kvůli přehlednosti. Je díky tomu možno lehce poznat, zda jsou příkazy psány pro hostitelský počítač, nebo jeden z virtuálních. Na konci čtenáře nabádám k tomu, aby dal studentům jako první úkol takovéto nastavování IP adres, to ale záleží již na učiteli na probrané látce.

10.1.5 Připojení sítě virtuálních počítačů do skutečné sítě

Tuto kapitolu jsem se rozhodl sepsat zejména kvůli tomu, abych předvedl, jak jednoduché je do virtuálních počítačů zapojovat více síťových karet či propojení virtuálního počítače s fyzickým. Propojení virtuálního počítače kombinuji s tvorbou virtuální sítě z předchozí kapitoly.

Předpokládám, že čtenář má znalosti týkající se překladu adres, proto v této kapitole nevysvětluji příkazy s programem iptables a podobné. Ze stejného důvodu ani nezmiňuji možnost použití MASQUERADE místo SNAT

a věřím, že čtenář využívající DNS server k připojení na internet bude vědět, co použít.

10.1.6 KVM ve scriptech

V této kapitole jsem čtenáři ukázal, jak je možné využít, když program umožňuje ovládání z příkazové řádky. Předvedl jsem dva ukázkové scripty a ke každému napsal, co se scriptem ovládá. Popis scriptu jsem vždy doplnil o návrhy, jak by mohl čtenář scripty ještě upravit. Věřím, že čtenáře budu motivovat k dalšímu přemýšlení nad možnostmi scriptů spojených s virtualizací, když ho takto navedu a on se do nějakých, z kraje třeba menších, změn pustí.

10.1.7 Koncem to nekončí

Závěrečnou kapitolou jsem se pokusil čtenáře motivovat k dalšímu hledání možností virtualizačního nástroje KVM.

10.2 Souhrn praktických tipů pro tvorbu virtuální desktopové infrastruktury

10.2.1 Virtuální desktopová infrastruktura

Touto kapitolou začínám část souhrnu praktických tipů zaměřenou na tvorbu učebny s podporou virtuální desktopové infrastruktury. V krátkosti popisuji, kdy je vhodné použít VDI a jaké má toto řešení výhody.

10.2.2 Instalace serveru

Popis instalace virtualizačního operačního systému na server. Vzhledem k jednoduchosti nainstalovat ESXi na server je tato kapitola krátká - instalátor serveru nenabízí mnoho voleb.

10.2.3 Ovládání serveru

Aby se čtenář rychleji dokázal zorientovat v prostředí vSphere Center, popisují prostředí tohoto programu pro vzdálenou správu serverů.

Při psaní této kapitoly vycházím z předpokladů, že se ještě pomocí vSphere Center neovládá žádný server.

V první polovině kapitoly se snažím, aby čtenář pochopil, kde se nachází jaký ovládací prvek, k čemu jej využít a jaké může mít hodnoty. Přehledně popisují jednotlivé záložky nastavení.

V druhé polovině kapitoly již předvádím postupy, které se využijí při tvorbě virtualizované učebny. Jedná se o připojení NFS disku, změnu adres ESXi serveru a instalaci virtuálního stroje metodou čisté instalace, nebo importem z předpřipraveného souboru. Obě možnosti instalace VM uvádím z toho důvodu, že při instalaci virtuální desktopové infrastruktury využijeme přesně tyto metody.

10.2.4 VDI-in-a-box

V této kapitole se dostávám k samotné tvorbě virtuální desktopové infrastruktury.

V první části se věnuji nastavení serverové části, tedy nainstalování a nastavení řešení VDI-in-a-box a přípravě obrazu, který bude použit jako šablona virtualizovaného desktopu ve virtuální desktopové infrastruktuře.

Na závěr uvádím několik rad, jak je možno připravit fyzický počítač, kterým se bude přistupovat k virtuální desktopové infrastruktuře. Nejen, že zmiňuji přistupování k VDI pomocí třeba nevýkonného PC, do kterého je mimo jiné nainstalován webový prohlížeč s doplňkem pro přístup k VDI-in-a-box, ale také přistupování k virtualizovaným desktopům pomocí tabletu s operačním systémem Android.

11 Závěr

V mé práci na téma Management virtualizace se mi podařilo dosáhnout stanovených cílů. Tyto cíle jsou:

1. Otestovat programy pro desktopovou virtualizaci (VMware workstation, Win Virtual PC, KVM).
2. Na základě testování vybrat jeden program, o kterém napíšu souhrn praktických tipů pro učitele sítí.
3. Otestovat řešení pro virtualizaci serverů a virtuální desktopovou infrastrukturu (platformy VMware a Citrix).
4. Na základě testování vybrat jednu platformu, o kterém napíšu souhrn praktických tipů na vytvoření virtualizované učebny.
5. Sepsat souhrn praktických tipů.

11.1 Použité zdroje

CITRIX [online]. 13.2.2012 [cit. 2012-12-19]. Citrix VDI-in-a-box. Dostupné z <http://support.citrix.com/proddocs/topic/vdi-51/>

CONNECT [online]. 2010 [cit. 2010-04-19]. Kategorie Virtualizace. Dostupné z <http://connect.zive.cz/serverova-virtualizace>

Debian – Balíčky. Debian [online]. 2012-4-03 [cit. 2012-04-17]. Dostupné z: <http://www.debian.org/distrib/packages>

KVM: Kernel Based Virtual Machine [online]. 2011-02-01 [cit. 2012-04-17]. Dostupné z: <http://www.linux-kvm.org/>

KVM I/O slowness on RHEL 6. Wwww.ilsistemista.net [online]. 2011-03-11 [cit. 2011-10-22]. Dostupné z: <http://www.ilsistemista.net/index.php/virtualization/11-kvm-io-slowness-on-rhel-6.html?limitstart=0>

Microsoft [online]. Praha : Microsoft, 2010 [cit. 2010-04-19]. Virtualizace. Dostupné z: <http://www.microsoft.com/cze/virtualizace/>

OLDANYGROUP. OldanyGroup [online]. 2005-2012 [cit. 2011-06-20]. Dostupné z: <http://www.oldanygroup.cz/>

OpenShift Express: Svobodný cloud od Red Hatu. Linuxexpres.cz [online]. 2011-06-28 [cit. 2012-01-26]. Dostupné z: <http://www.linuxexpres.cz/business/openshift-express-svobodny-cloud-od-red-hatu>

Ukončení podpory: Podpora Windows XP SP3 a Office 2003 končí 8. dubna 2014. MICROSOFT. Microsoft.com [online]. 2012 [cit. 2012-04-18]. Dostupné z: <http://www.microsoft.com/cze/ukoncenipodpory/>

ÚVT MU [online]. 13.4.2010 [cit. 2010-04-19]. Virtualizace. Dostupné z <http://ics.muni.cz/zpravodaj/articles/545.html>

Virtualization with Virt-Manager and KVM in Ubuntu 8.04. Tombuntu.com [online]. 2008 [cit. 2011-03-28]. Dostupné z: <http://tombuntu.com/index.php/2008/04/14/virtualization-with-virt-manager-and-kvm-in-ubuntu-804/>

Windows Virtual PC: Domovská stránka: Režim Windows XP a technologie Windows Virtual PC. MICROSOFT. Microsoft.com [online]. 2011 [cit. 2012-04-18]. Dostupné z: <http://www.microsoft.com/cze/windows/virtual-pc/default.aspx>

XFCE: Seriál o XFCE. HORONČOK, Miroslav. Linuxexpres.cz [online]. 2008 [cit. 2011-03-11]. Dostupné z: <http://www.linuxexpres.cz/software/xfce>

11.2 Přílohy práce

K práci přikládám:

- Tištěnou verzi „Souhrn praktických tipů nejen k výuce sítí podpořené virtualizací“ a „Souhrn praktických tipů pro virtuální učebnu“
- CD s elektronickou verzí práce a souhrnu praktických tipů (pdf)

Souhrn praktických tipů nejen k výuce sítí podpořené virtualizací

Lukáš Komárek

1 Virtualizace při výuce sítí

1.1 Popis problému

Praktická výuka sítí ve škole předpokládá dobré a většinou i nákladné vybavení. To je zejména kabelové propojení počítačů, routerů, switchů a serverů, samotné počítače a servery, jiné síťové prvky a v neposlední řadě také prostory. Vzhledem k tomu, že pokud bude chtít více studentů nastavovat servery ve stejný čas, bude zapotřebí i více serverů, než jen jeden.

A co teprve časová náročnost na vrácení serveru (i jiných síťových prvků) do původní podoby, aby se další student nemusel trápit s tím, co předchozí student přenastavil? Pokud je to vůbec v našich silách, zajisté to bude zbytečně dlouho trvat a bude potřeba vynaložit nemalé úsilí pro objevení změn od původního nastavení.

Představa takovéto výbavy a časové náročnosti může nejednu školu od výuky sítí odradit. Při použití virtualizace nám většina těchto problémů odpadne a pro jednoho žáka bude postačovat pouze jeden výkonnější počítač.

Pro použití virtualizace ve výuce sítí připadá v úvahu virtualizační nástroj KVM na Linuxovém operačním systému.

1.2 Návrh řešení na platformě Linux

1.2.1 Hardwarové vybavení počítače

Hardwarové parametry fyzického počítače volíme podle toho, kolik virtuálních počítačů budeme chtít na daném počítači spouštět a jaké operační systémy to budou. Zjednodušeně se dá říci, že operační systém s virtualizačními nástroji využije 10-15% výkonu fyzického počítače, o zbytek se mohou podělit virtuální stroje. Procesor je srdce počítače a v tomto případě to platí několikanásobně, protože budeme spouštět více virtuálních počítačů, je tedy vhodné vybírat výkonné procesory a vzhledem k adresaci více RAM volit 64b architekturu. Pro naše účely je důležité, abychom měli v počítači procesor s podporou hardwarové virtualizace. U procesorů Intel se technologie nazývá Intel Virtualization Technology (Intel VT), u procesorů AMD se technologie nazývá AMD-V. Minimální kapacitu paměti RAM a pevného disku zvolíme tak, že sečteme, kolik jí potřebuje samotný hypervizor a jednotlivé virtuální stroje (u kapacity pevného disku záleží na použitém formátu obrazů - je možné zvolit kapacitu nižší). Základní deska musí být osazena patičkami pro vybraný procesor a umožňovat připojení dostatečného množství paměti RAM. Pokud budeme chtít, aby na každém počítači bylo nainstalováno 15 obrazů disků s linuxovým operačním systémem a 3 obrazy se systémem Microsoft Windows, z nichž se vždy 7 obrazů zazálohují a spustí, budeme potřebovat zhruba 360 GB disk, 8 GB RAM, dostatečně rychlý procesor a odpovídající základní desku. Udané hodnoty jsou minimální a spíše orientační - vždy je vhodné uvážit jaké virtuální systémy se budou spouštět a počítač sestavovat tak, aby byl výkon vyšší.

1.2.2 Softwarové vybavení počítače

Distribuci operačního systému doporučuji volit podle zkušeností s ní a nároků na hardware. Pokud nemáte zkušenosti s žádnou distribucí, nebo se nemůžete rozhodnout, doporučuji Debian s grafickým rozhraním XFCE. Distribuci ke

stažení i instalační příručku najdete například na stránkách www.debian.org. V textu tedy budu uvádět příkazy určené pro operační systém Debian, v ostatních distribucích se kód může mírně lišit.

Když máme nainstalovaný operační systém, je ještě potřeba doinstalovat kompletní virtualizační řešení KVM. To uděláme příkazem:

```
apt-get install kvm
```

A pro jistotu zkontrolujeme, zda máme v jádře zavedeny moduly `kvm.ko` a `kvm-intel.ko` nebo `kvm-amd.ko` (podle procesoru)

```
lsmod | grep kvm
```

Pokud nám terminál vypíše dva řádky, které budou začínat jmény výše uvedených modulů, máme systém připraven pro virtualizaci.

1.2.3 Instalace virtuálního operačního systému

Virtuální disk Abychom mohli nainstalovat virtuální operační systém, potřebujeme virtuální disk (tzv. image disku), na který se vše nainstaluje. Virtuální disk bude na fyzickém disku reprezentován jedním souborem. Tvorbu disku provedeme programem `kvm-img` spuštěným z příkazové řádky v následující podobě.

```
kvm-img create -f qcow2 -o preallocation=metadata virtualniDisk.img 20G
```

Příkazem `create` říkáme programu `kvm-img`, že bude vytvářet nový image disku.

Parametrem `-f` nastavíme formát virtuálního disku. Tímto parametrem neformátujeme oddíly na disku, ale volíme formát image disku - jednoho souboru. Výběrem formátu můžeme ovlivnit, jak bude virtuální disk rychlý, nebo třeba kolik bude zabírat místa na disku fyzickém. Formát `qcow2` volím z důvodu, že mi virtuální disk bude zabírat na disku pouze tolik místa, kolik bude virtuální disk obsahovat dat. Pokud bych použil např. formát `raw`, zabíral by obraz disku tolik místa, jak velkou kapacitu by měl, ovšem byl by o něco rychlejší, jelikož by se systém nemusel zatěžovat (de)kompresí. To znamená, že virtuální disk o zvolené velikosti 20GB s nově nainstalovaným linuxovým systémem bude zabírat na fyzickém disku zhruba 800MB (dle toho, co vše instalujeme) při formátu `qcow2` a 20GB při formátu `raw`.

Načítání dat z obrazů formátu `qcow2` výrazně zrychlíme, když aktivujeme prealokaci `-o preallocation=metadata`, dále už jen zvolíme název a velikost virtuálního disku. Velikost můžeme udat v jednotkách b, K, M, G, T.

Instalace systému Instalace operačního systému probíhá skoro stejně, jako na fyzickém stroji. Nejprve je nutno získat instalační cd, nebo lépe iso obraz instalačního cd - instalace z iso souboru proběhne rychleji, než z cd. U většiny linuxových distribucí se nechá stáhnout obraz s aktuálním systémem z internetu. V případě, že si chceme udělat iso soubor z CD, můžeme to udělat z příkazové řádky např. programem `dcfldd` v podobě

```
dcfldd if=/dev/cdrom of=obraz_disku.iso
```

Když máme nachystané obrazy disků, můžeme začít instalovat virtuální počítač.

```
kvm -hda virtualni_disk.img -cdrom instalacni_cd.iso -boot d -m 1024M
```

Přepínači `-hda` a `-cdrom` určíme soubory s virtuálním a instalačním diskem. `-boot d` říká, že se má systém zavádět z cdromu, není ovšem povinný, pokud jde o instalaci na prázdný virtuální disk (nový/bez zaváděcího záznamu).

Přepínačem `-m` zvolíme, s jak velkou pamětí RAM virtuální počítač spustíme. Hodnoty můžeme udávat v M nebo G, pokud neudáme nic, využije se 128MB paměti.

Po potvrzení příkazu se spustí virtuální počítač s naboootovaným instalačním diskem. V instalaci postupujeme stejně, jako bychom instalovali operační systém na fyzický počítač.

Co s nainstalovaným systémem? Hned po instalaci je vhodné si udělat zálohu virtuálního disku. Např takto:

```
cp virtualni_disk.img debian_6_po_instalaci.img
```

Na této záloze nebudeme provádět žádné změny, využijeme jí pouze v případě, když budeme chtít pracovat na čerstvě nainstalovaném systému, a to tak, že si ji překopírujeme. Tímto krokem se vyhneme opakované instalaci a ušetříme si čas.

Je mnoho možností, jak uchovávat a spouštět virtuální stroje na více počítačích - řekněme ve školní třídě. Nabízí se třeba možnost rozkopírování virtuálních disků do každého počítače, kde si každý student nastaví virtuální počítač(e) podle sebe a jejich nastavení bude uchovávat na svém disku, nebo můžeme image s virtuálními počítači nechat pouze na síťovém disku a spouštět je z něho. Určitě nebudeme chtít, aby se měnil virtuální systém na síťovém disku, proto k němu budeme přistupovat programem KVM jedině s přepínačem `-snapshot`, který zajistí, aby se všechny změny ukládaly jen jako dočasné soubory a nepřepisovaly soubor s virtuálním diskem. Pro jistotu je dobré mít síťový disk, ke kterému přistupuje více uživatelů s právy pouze

pro čtení - tam se přepínač -snapshot použít musí. Scripty pro spouštění více počítačů s nastavením pro různé situace si předvedeme později.

Zkusme si teď spustit virtuální počítač, který jsme si nainstalovali.

```
kvm -hda virtualni_disk.img -m 256M
```

Příkaz jistě není potřeba vysvětlovat, vše již známe.

1.2.4 Propojení více virtuálních počítačů do virtuální sítě

Předvedu dvě možnosti, jak je možné připojit více virtuálních počítačů do virtuální sítě. První bude pomocí TCP socketu, druhá virtuální síť bude tvořena multicastem. Síť tvořená TCP socketem má větší latenci, zato vyšší přenosovou rychlost. Naopak virtuální síť tvořená multicastem má velmi nízkou latenci a nižší rychlost. Pokud budete chtít mít v síti virtuální počítače s nějakým OS od Microsoftu, volte první možnost. Multicastové propojení spojí virtuální počítače na stejném multicastu a portu na všech hostitelích v síti.

Pomocí TCP socketu První možností je připojení pomocí TCP socketu, kdy jeden virtuální počítač naslouchá na daném TCP portu a ostatní počítače se na tento port připojují přes loopback hostitelského PC.

První virtuální stroj spustíme příkazem (VM0)

```
kvm -hda linux.img -m 256M -snapshot \  
-net nic,macaddr=52:54:00:11:22:33,model=virtio \  
-net socket,listen=:1234
```

Další stroj tímto příkazem (VM1)

```
kvm -hda linux.img -m 256M -snapshot \  
-net nic,macaddr=52:54:00:33:22:11,model=virtio\  
-net socket,connect=127.0.0.1:1234
```

Parametr `-snapshot` zde je kvůli tomu, že spouštíme počítače ze stejného virtuálního disku. Pokud budeme spouštět najednou rozdílné počítače, můžeme `-snapshot` vynechat.

`-net nic` nastavuje síťovou kartu, se kterou bude virtuální počítač spuštěn, v tomto případě nám stačí nastavit jen její mac adresu. Model `virtio` nastavuji proto, že je s ní rychlejší datový přenos, než s emulovanou kartou `RTL8139`, která by byla ve virtuálním počítači, kdybychom model nezadali. U některých operačních systémů je potřeba doinstalovat kartě `virtio` ovladače - odkaz ke stažení je na oficiálních stránkách KVM. Při spouštění více virtuálních počítačů dbejte na to, aby dva počítače neměly stejnou mac adresu.

U prvního počítače nesmíme zapomenout nastavit port, na kterém bude naslouchat síťovému provozu, zde `1234`. Další počítače spouštíme obdobně, ale necháme je připojovat přes `loopback` fyzického počítače na stejný port, jako jsme nastavili u prvního stroje. `Loopback` je většinou nastaven na `127.0.0.1`, případně se nechá zjistit programem `ip` - síťové rozhraní `lo` nebo napsáním příkazu

```
traceroute localhost
```

atp..

Pomocí multicastu Každému počítači dáme síťovou kartu naslouchající stejnému UDP multicastu.

Spuštění prvního počítače (VM0):

```
kvm -hda linux.img -snapshot -m 256M \  
-net nic,macaddr=52:54:00:12:34:56 \  
-net socket,mcast=230.0.0.1:1234
```

Spuštění dalšího počítače (VM1):

```
kvm -hda linux.img -snapshot -m 256M \  
-net nic,macaddr=52:54:00:65:43:21 \  
-net socket,mcast=230.0.0.1:1234
```

Oba příkazy jsou v podstatě stejné, jen s rozdílnou mac adresou. Pomocí `-net socket,mcast=...` řekneme síťové kartě, k jaké adrese a portu multicastu se má připojit.

Funkční síť Ať už jste si vybrali jakoukoli možnost spojení počítačů do virtuální sítě, na spuštěných virtuálních počítačích nastavte síťovým kartám IP adresy ve stejné síti a přesvědčte se, zda na sebe vidí. Jistě říkám zbytečně, že do příkazových řádek virtuálních počítačů budete psát něco jako toto:

VM0:

```
ip a add 10.0.0.1/24 brd + dev eth0  
ip link set up dev eth0
```

VM1:

```
ip a add 10.0.0.2/24 brd + dev eth0  
ip link set up dev eth0
```

VM0:

```
ping 10.0.0.2
```

VM1:

```
ping 10.0.0.1
```

Nastavování IP adres virtuálním počítačům můžete využít jako první úkol v hodině sítě podpořené virtualizací.

1.2.5 Připojení sítě virtuálních počítačů do skutečné sítě

Pro případ, že budeme chtít aby virtuální počítače komunikovaly s počítači mimo virtuální síť, využijeme hostitelský počítač jako router. Skutečnou síť, do které je připojen hostitelský pc skryjeme za NAT a na onom PC vytvoříme virtuální síťové rozhraní, ke kterému bude připojen minimálně jeden virtuální počítač. Na hostitelském počítači musí být nainstalován balík `uml-utilities`.

Dle předchozího návodu máme vytvořenou síť virtuálních počítačů, kterou chceme připojit do skutečné sítě. Proto spustíme jeden virtuální počítač, který bude zastávat funkci routeru. Tento virtuální počítač bude připojen jedním síťovým rozhraním ke skutečné síti a druhým do virtuální sítě.

Řekněme, že hostitelský počítač je připojen k síti, případně k internetu, síťovou kartou eth0 s pevnou IP adresou 192.168.100.5/24. Nastavme na něm SNAT a aktivujme IP forwarding:

```
iptables -t nat -A POSTROUTING -o eth0 -j SNAT --to 192.168.100.5
echo "1" > /proc/sys/net/ipv4/ip_forward
```

Vytvoříme virtuální síťové rozhraní vnic0 a přiřadíme mu IP adresu z jiné sítě:

```
tunctl -t vnic0
ip a add 192.168.101.1/24 brd + dev vnic0
ip link set up dev vnic0
```

Nyní spustíme virtuální počítač VM2 a dáme mu dvě síťové karty. Jednu jeho síťovou kartu připojíme k virtuální “kartě” vnic0, druhou připravíme na spojení s virtuální sítí tvořenou multicastem:

```
kvm -hda virtual.img -m 256M -snapshot \
-net nic,macaddr=52:54:00:12:12:12 \
-net tap,ifname=vnic0,script=no \
-net nic,macaddr=52:54:00:13:13:13,vlan=1 \
-net socket,mcast=230.0.0.1:1234,vlan=1
```

Nyní nastavíme na VM2 IP adresu v síti s vnic0, výchozí brána bude nastavena na IP adresu virtuálního rozhraní hostitelského počítače. Pokud budeme chtít přistupovat k internetu, zadáme i DNS servery (musí být ,v našem případě, připojen hostitelský počítač síťovou kartou eth0):

VM2:

```
ip a add 192.168.101.2/24 brd + dev eth0
ip link set up dev eth0
ip route add default via 192.168.101.1
echo "nameserver ip.adresa.dns.serveru" >> /etc/resolv.conf
# Druhé síťové kartě dáme adresu v síti s ostatními virtuálními
# počítači a nastavíme routování
ip a add 10.0.0.3/24 brd + dev eth1
ip link set up dev eth1
iptables -t nat -A POSTROUTING -o eth0 -j SNAT --to 192.168.101.2
echo "1" > /proc/sys/net/ipv4/ip_forward
```

Počítačům ve virtuální síti nastavíme jako výchozí bránu VM2, DNS a je hotovo.

1.2.6 KVM ve scriptech

To, že se KVM ovládá přes BASH, je možné využít pro spouštění mnoha počítačů a tvorby sítě pomocí scriptů. Tvorba scriptů je neomezená a nechá se udělat script v podstatě na vše. Zde vám předvedu pouze pár ukázek, které můžete využít pro urychlení vaší práce.

Automatické spuštění několika virtuálních strojů s náhodnou mac adresou:

```
#!/bin/bash

#cesta k síťovému disku
sitovy_disk=/media/nfs/

#názvy souborů s virtuálními počítači
prvni=centOS.img
druhy=gentoo.img
treti=openBSD.img

#spuštění počítačů
for obraz in $prvni $druhy $treti
do
kvm -hda $sitovy_disk$obraz -snapshot -m 256M \
-net nic,macaddr=52:54:00:54:${RANDOM:0:2}:${RANDOM:0:1}${RANDOM:0:1} \
-net socket,mcast=230.0.0.1:1234&
sleep 2s
done
```

Ve scriptu je potřeba doplnit cestu k síťovému disku a názvy souborů s virtuálními disky. Při spouštění scriptu rovnou ze síťového disku(= aktuální adresář bude v tomto případě /media/nfs/) je možné místo cesty zadat ./ atp.. Potom je vhodný k rozeslání zpráv na spuštění třech virtuálních počítačů na každém hostiteli. Pokud jsou hostitelské počítače v jedné síti, budou propojeny do sítě virtuální počítače ze všech hostitelů.

Náhodná mac adresa není ošetřena na 100%, v některých případech by se mohlo stát, že se vygeneruje stejná adresa. Vyloženě špatná adresa, kvůli které by se jeden virtuální počítač nespustil vůbec, se také může vygenerovat (kvůli \${RANDOM:0:2} - můžete vyměnit za dvě jedničky, jako je tomu vedle), ovšem tato pravděpodobnost je velmi nízká.

Script není problém předělat na spouštění více počítačů pouhým zavedením proměnné s názvem virtuálního disku a tuto proměnnou dopsat do cyklu for.

V případě nevyplnění názvu virtuálního disku se nic špatného nestane, prostě se jeden virtuální počítač nespustí.

Script na připojení sítě k internetu, různé typy OS

```
#!/bin/bash

#cesta k síťovému disku
sitovy_disk=/media/nfs/

#název souboru routeru a jeho RAM v MB
router=debian.img
router_ram=128

#1. skupina klientů a jejich RAM v MB
klient1=windows7.img
klient2=windowsXP.img
moc_ram=768

#2. skupina klientů a jejich RAM v MB
klient3=xubuntu.img
klient4=debian.img
malo_ram=256

#Síťová karta v internetu
nic=eth0

#Nastavení sítě
```

```

tunctl -t vnic0
ip a add 10.10.13.1/24 brd + dev vnic0
ip link set up dev vnic0
iptables -t nat -A POSTROUTING -o $nic -j MASQUERADE
echo 1 > /proc/sys/net/ipv4/ip_forward

#Spouštění routeru
kvm -hda $cesta$router -m ${router_ram}M -snapshot \
-net nic,macaddr=52:54:00:12:12:12 \
-net tap,ifname=vnic0,script=no \
-net nic,macaddr=52:54:00:13:bb:13,vlan=1 \
-net socket,listen=:1234,vlan=1&

#spouštění klientů
for obraz in $klient1 $klient2 do kvm -hda $cesta$obraz -m ${moc_ram}M -snaps
-net nic,macaddr=52:54:00:33:22:${RANDOM:0:1}${RANDOM:0:1} \
-net socket,connect=127.0.0.1:1234&
done

for obraz in $klient3 $klient4
do
kvm -hda $cesta$obraz -m ${malo_ram}M -snapshot \
-net nic,macaddr=52:54:00:33:28:${RANDOM:0:1}${RANDOM:0:1} \
-net socket,connect=127.0.0.1:1234&
done

```

Může se stát, že budete potřebovat spouštět různorodé systémy, které mají jiné nároky. Proto jsem vám připravil script, kterým je možno spustit dva druhy operačních systémů a každému typu přidělit jinou velikost RAM. Nemusíte se omezovat jen na paměti, script se nechá jednoduše upravit i na jiné rozdílné parametry - je to pouze na vás. Stejně jako u předchozího scriptu, i zde doplníte názvy souborů atp. Ve scriptu se spouští i jeden router, který

bude připojen k internetu, proto musíte vyplnit, jaká fyzická síťová karta bude připojena k internetu. Ve spuštěném routeru zadat výchozí bránu v síti 10.10.13.1 a zprovoznit routování pro ostatní počítače v síti - tedy ty, jenž jsou ve scriptu označeni jako klient1-4.

Dovedu si představit použití tohoto scriptu při cvičení třeba na nastavování Active Directory či na spojování různorodých systémů pomocí Samby.

1.2.7 Koncem to nekončí

Kapitolou se scripty bych mé tipy ukončil, ovšem to neznamená, že KVM nemá více možností. Tento úžasný virtualizační nástroj má parametry, o kterých jsem se zde nezmiňoval. Zkuste použít manuálové stránky a něco se přiučit z nich. Už jste slyšeli o parametru -nographic a VNC připojení?

Souhrn praktických tipů pro tvorbu virtualizované učebny

Lukáš Komárek

1 Virtuální desktopová infrastruktura

Může se stát, že budeme chtít provozovat software, který vyžaduje větší výkon na počítačích, které požadovaný výkon nejsou schopny poskytnout. Máme možnost pomocí virtualizace toto zajistit, nebo bude nutné dokupovat nové a výkonné počítače?

Tento problém můžeme vyřešit pomocí virtuální desktopové infrastruktury, kdy starší počítače budou v rolích klientů, kteří pouze zobrazují výstup virtuálního počítače spuštěného na serveru.

1.1 Instalace serveru

Jako hypervizor nám poslouží server s operačním systémem VMware ESXi, na který nainstalujeme rozšíření vdi-in-a-box. Instalace operačního systému je jednoduchá, stačí zavést systém z bootovatelného disku a nechat se vést instalátorem.

Po nabofování instalátoru ESXi z instalačního média instalátor zjistí konfiguraci serveru a zeptá se, na jaký disk hodláme systém instalovat - vybereme tedy požadovaný disk a potvrdíme. Poté nabídne rozložení klávesnice - česká klávesnice nechybí (czech). V této chvíli již můžeme instalovat VMware ESXi na náš server a při tom si zalít vodou šálek zeleného čaje.

Po instalaci a restartu serveru je na obrazovce vidět, jakou má server IP adresu. Pokud jí hodláme změnit, stiskneme klávesu f2, vložíme login a zavítáme do Configure Management Network a dále Network Adapters, kde již požadované nastavení provedeme. Nyní již můžeme server ovládat vzdáleně.

1.2 Ovládání Serveru

Server se ovládá pomocí programu vSphere Center od společnosti VMware. Instalace programu musí být provedena na počítač s OS Windows XP a novějším, je jednoduchá, takže nepotřebuje komentář. Po spuštění programu zadáme uživatelské jméno a heslo (jméno root, heslo jsme zadávali při instalaci) a připojíme se.

1.2.1 Popis prostředí vSphere Center

V levém pruhu je server, ke kterému jsme připojení (ip adresa tohoto serveru), a po rozkliknutí tohoto serveru se pod ním zobrazí názvy nainstalovaných virtuálních počítačů (pokud pracujete na čisté instalaci serveru, logicky se nezobrazí žádný virtuální stroj). V pravé části je prostor pro kontrolování stavu a správu serveru či virtuálních počítačů, které vybereme v levé části. Jsou zde záložky Summary, Resource Allocation, Performance, Events a Permissions s

tím rozdílem, že u serveru ještě navíc záložky Virtual Machines, Configuration, Local Users & Groups a u virtuálních počítačů záložka Console. Prvky v záložkách, které jsou u serveru i u VM jsou podobné, obvykle je zde logický rozdíl pouze v tom, že u VM nastavujeme či diagnostikujeme hodnoty vztahující se k jednotlivému virtuálnímu stroji, kdežto u serveru jde o hodnoty vztahující se ke všem VM na serveru běžícím (uloženým) a ještě k tomu k samotnému fyzickému serveru. Nyní si stručně popíšeme jednotlivé záložky:

Summary Celkový souhrn většiny podstatných informací. Můžeme zde vidět například kapacitu zdrojů - volná i využitá operační paměť, CPU, stav stroje (vypnutý/zapnutý), úložných zařízení, informace o hardwaru (fyzickém i virtuálním) a několik rychlých příkazů, jako vypnout, zapnout, uspat.

Virtual Machines Výpis VM uložených na serveru, zobrazení využívání prostředků jednotlivými VM, jejich stavu.

Resource Allocation Pod touto záložkou je možno nastavit, kolik prostředků dovolíme virtuálnímu počítači využívat. Nastavují se zde hodnoty pro CPU, operační paměť, a úložný prostor.

Performance Zobrazení využití systémových zdrojů v grafu s časovou osou do minulosti. Hodnoty, které chceme zobrazovat vybereme v kolonce Switch to.

Configuration Do záložky Configuration zavítáme, když budeme chtít spravovat server. Je možné přidávat disková zařízení, měnit nastavení síťových adaptérů, vytvářet virtuální switche, automaticky spouštět VM po startu serveru, upravovat čas, vytvořit firewall či je možné třeba nastavit, kolik milisekund se má čekat, když je zpožděná reakce disku, než se napíše varování do logu. Nastavit se zde nechá skutečně mnoho, doporučuji záložku Configuration pro představu alespoň projít.

Local Users & Groups Zde je možno přidat, smazat či upravit uživatele a skupiny, změnit hesla či je propojit mezi sebou (uživatele a skupiny).

Events Toto je log událostí. Je vhodné sem zavítat, když si nejsme jisti, co se se serverem nebo s VM dělo. Na každou událost je možno kliknout a tím zobrazit její podrobnější obsah.

Console V consoli je vidět obrazovka virtuálního počítače, využívá se v případě, že se k počítači nepřistupuje vzdáleně, nebo při instalaci. Práce přes consoli není příliš svižná ani pohodlná.

Permissions Zde se nastavuje oprávnění uživatelům nebo skupinám. Práva jsou rozdělena na tři kategorie read-only, administrator, no-access.

Ještě před instalováním virtuální desktopové infrastruktury se podíváme na nastavení několika základních věcí na serveru ESXi za pomoci vSphere Clienta.

Připojení síťového NFS disku Pod záložkou Configuration v sekci hardware zvolíme storage a volbou Add Storage... vyvoláme průvodce pro přidání nového disku. Jako Storage Type vybereme Network File System a klikneme na Next. Do kolonky server vepíšeme ip adresu serveru a jako Folder napíšeme umístění sdíleného adresáře na serveru. Zaškrtnutím check boxu Mount NFS Read Only připojíme disk pouze pro čtení. Nakonec do kolonky Datastore Name napíšeme jméno disku, které se bude zobrazovat ve výpisu disků. Může to vypadat například takto:

Server: 10.0.0.101

Folder: /mnt/iso

Datastore Name: ISO Obrazy

Změna IP adresy ESXi serveru IP adresa serveru může být nastavena pevně, nebo je možné ji získávat od DHCP serveru v síti. Nastavení IP adresy se provádí opět pod záložkou Configuration v sekci Hardware, tentokrát volbou Networking. U nastavení virtuálního switchu klikneme na možnost Properties... a tlačítkem Edit... upravíme nastavení Management Network. Toto nastavení není nutné provádět pouze z vSphere Clienta, lze to udělat i přímo na serveru bez vzdáleného připojení(f2, login, Configure Management Network, Network Adapters).

Instalace virtuálního počítače Jak nainstalovat nový virtuální počítač do VMware ESXi pomocí vSphere client? Máme dvě možnosti - buď importem nainstalovaného virtuálního stroje ve formátu OVF/OVA(pravda, není to instalace nového OS, ale již někým předpřipravený virtuální počítač a jeho pouhý import do ESXi, ovšem i tato metoda se využívá), nebo zvolením virtuálního hardwaru a čistou instalací OS z vlastního instalačního média.

Čistá instalace Odkaz na vytvoření nové VM je v vSphere clientovi hned několikrát, najít jej lze například kliknutím v levém sloupci pravým tlačítkem myši a zde zvolit New Virtual Machine... Na první obrazovce instalačního průvodce vybereme Typical, na další napíšeme jméno nového virtuálního stroje. Na stránce Storage zvolíme úložné zařízení, na kterém chceme uchovávat soubory tohoto virtuálního počítače. Stránkou Guest Operating System nám instalátor nabízí volbu virtualizovaného operačního systému. Nenabízí přímo instalační médium, ale pomůže přednastavit virtualizovaný hardware tak, aby danému operačnímu systému vyhovoval. Pokud si v široké nabídce OS nevybereme, zvolíme možnost

Other, Other(32/64). Na další straně přiřadíme počítači virtuální síťové karty. Desktopovému operačnímu systému bude stačit pravděpodobně jedna, server jich, dle role, využije možná více.

Nyní se dostáváme k vytváření virtuálního disku pro VM, kde je zásadní velikost diskového prostoru a také způsob reprezentování virtuálního disku v datovém úložišti. Volbami Thick Provision je obsazemé místo na fyzickém disku dáno velikostí virtuálního disku, naproti tomu volba Thin Provision zajistí, že virtuální disk zabere tolik místa, kolik dat do něj zapíšeme. Logicky vyplývá, že disk Thin P. bude o něco pomalejší, ale nebude zabírat zbytečné místo v datovém úložišti. Výběr metody nechám na vás, záleží na prioritách.

Na poslední stránce vidíme souhrn informací o nově vytvářeném virtuálním stroji, pod kterým je možnost Edit the virtual machine settings before completion. Tuto možnost zaškrtneme, abychom ještě mohli přidat informace o instalačním médiu. Po stisknutí tlačítka Continue se zobrazí okno s nastavením virtuálního hardwaru, nás zajímá možnost New CD/DVD, kde vybereme mezi možnostmi Client device, Host device nebo Datastore iso file. První možnost připojí do VM cd z mechaniky počítače, na kterém je spuštěn vSphere Client, druhá možnost připojí cd z mechaniky serveru ESXi a poslední možností připojíme do VM ISO obraz z datového úložiště, třeba ze sítě tato možnost je dle mého názoru nejjednodušší. Ještě zaškrtneme v horní části okna možnost Connect at power on, aby se instalační disk připojil hned při startu VM a Źukneme na Finish.

Nyní se v levém sloupci zobrazil nový virtuální počítač s názvem, jaký jsme mu zvolili na začátku. Spuštění VM provedeme tak, že na tento počítač klikneme a v horní části klikneme na ikonu Power on (zelený trojúhelník, neboli play). Teď již pokračujeme v instalaci přes záložku Console, kde se instalace neliší od instalace OS na fyzický hardware. Po instalaci nezapomeneme odpojit instalační cd-rom v okně Edit settings, které vyvoláme třeba ze záložky Summary.

Import VM z OVF/OVA souboru Některé OS se nešší pouze instalačními disky či jejich ISO obrazy, ale třeba již nainstalované ve formátu OVA nebo OVF, například VDI-in-a-box, kterému se budeme věnovat o několik řádek níže. Prakticky to probíhá tak, že si z internetu stáhneme soubor, v vSc jdeme do nabídky file a vybereme možnost Deploy OVF Template... a zadáme cestu k tomuto souboru. Na další stránce vidíme, kolik místa tento počítač zabere v úložišti dat, pokud použijeme Thin nebo Thick Provision. Pokračujeme tlačítkem Next a zadáme název VM, případně ponecháme nynější. Poté si vybereme, jakou chceme využít možnost ukládání dat, vybereme dle volného místa v datovém úložišti a pokračujeme na další stranu, abychom si mohli přečíst informace o nově přidávaném virtuálním počítači do ESXi serveru. Pokud je vše

v pořádku, stačí stisknout Finish a spustit samotný import VM. Po pár minutách vyskočí hláška o úspěšném přidání VM, tu zavřeme a v levém sloupci máme o jeden počítač více, se kterým můžeme pracovat jako s ostatními.

1.3 VDI-in-a-box

1.3.1 Server

Virtuální desktopovou infrastrukturu bude zajišťovat nový virtuální počítač běžící na serveru ESXi. Toto řešení se nazývá vdi-in-a-box a je od společnosti Citrix. Počítač v podobě souboru ve formátu .ova si stáhneme ze stránek CITRIXu a přes vSphere Center naimportujeme do serveru. Vdi-in-a-box se spravuje přes internetový prohlížeč. Adresa do tohoto prostředí je

`https://<ip_adresa_virtualniho_pc_VDI-in-a-box>/admin`

Po zadání přihlašovacích údajů (jméno: vdiadmin, heslo: kaviza) jsme v prostředí vdi-in-a-box vyzváni k tvorbě nové virtuální desktopové infrastruktury. Nejdříve je nutné zadat informace o ESXi serveru, na kterém VDI poběží, zadáme tedy IP adresu, uživatelské jméno a heslo k fyzickému serveru, poté vybereme úložiště, na kterém budou uchovávány virtuální počítače a síť, kterou budou využívat. Na další obrazovce zvolíme možnost vytvoření nového gridu a budeme pokračovat na nastavení domény, kde budeme chtít, aby se naše desktopová infrastruktura připojovala do již stávající Active Directory domény. Proto vyplníme IP adresu řadiče domény, název domény a přístupové údaje k ní. Aby bylo možné zapojovat počítače z této infrastruktury do sítě, musíme mít rezervované IP adresy, které počítačům budeme moci přiřadit. Pokud to tak máme, potvrdíme to na další stránce a pustíme se do vytváření obrazu virtuálního počítače.

V ESXi serveru vytvoříme nový virtuální stroj a do něho nainstalujeme operační systém Windows 7. Tento obraz ještě není zcela připraven pro použití ve virtuální desktopové infrastruktuře, je nutné do něho nainstalovat VMware Tools, povolit vzdálený přístup a některé porty ve firewallu, vložit licenční klíč, připojit do domény a přidat jednu hodnotu do registrů.

Nejdříve nainstalujeme VMware tools, aby se s VM lépe pracovalo - v vSphere pravým tlačítkem myši klikněte na spuštěný VM, kde hodláte instalovat VMware tools a v nabídce guest vyberte Install/Upgrade VMware Tools, potvrďte, že máte virtuální stroj spuštěný tlačítkem OK a ve virtualizovaném systému instalujte program z CD (možná se spustí automaticky), po několikerém stisknutí tlačítka Next a restartu VM bude hotovo a VMWare Tools budou nainstalovány. Dále si otevřete nastavení brány firewall a povolte porty 139, 1494, 2598, 3389 a 8080, případně bránu úplně vypněte, pokud v cestě do internetu stojí jiná brána firewall (stále pracujete ve virtuálním stroji s W7). Vzdálený přístup

povolíte, když pravým tlačítkem myši kliknete na Tento počítač a ve vlastnostech najdete nastavení vzdáleného přístupu a tam povolíte vzdálenou plochu a přidáte oprávnění pro doménové uživatele (například přidat oprávnění skupině VDI_UZIVATELE umístěné v řadiči domény a každého uživatele, kterému budeme chtít poskytnout oprávnění pro připojování k VDI do této skupiny přidat). Do systému vložte platný licenční klíč získaný z některého z multilicenčního programu od společnosti Microsoft pro požadovaný počet stanic. Připojení do domény snad nemusím vysvětlovat.

Pomocí Spustit, z nabídky Start ve virtuálním počítači, spustíme regedit, program na úpravu registrů. S jeho pomocí přejdeme do

```
HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\policies\system
```

a tam vytvoříme novou DWORD hodnotu s názvem

```
LocalAccountTokenFilterPolicy
```

s údajem hodnoty 1. Když máme virtuální počítač připravený, necháme ho zaplýt a pokračujeme dále v prostředí vdi-in-a-box, kde vybereme jméno našeho obrazu, napíšeme jméno pro nový obraz a popíšeme ho. Pokud je za jménem obrazu napsáno not importable, zřejmě jsme neudělali všechny potřebné věci při přípravě obrazu nebo VM neběží, proto znovu zkontrolujeme nastavení, případně znovuspustíme VM s Windows 7.

Dále budeme vyzváni k zadání přístupových údajů k administrátorskému účtu počítače - vybereme Local user.. a zadáme přístupové údaje k administrátorskému účtu ve VM. Nyní se začne připravovat nový obraz pro VDI a celá operace bude chvíli trvat. Mohlo by se stát, že by operace skončila chybou instalace programu desktop agent, v tom případě obvykle stačí restartovat VM a zkusit operaci opakovat, pokud to nepomůže, postupujte dle instrukcí na obrazovce a program do VM nainstalujte ručně. Tato situace pravděpodobně nenastane, ale i když ano, povede se jí lehce vyřešit. Nyní stiskněte tlačítko Connect a zkuste se připojit zadáním uživatelských údajů doménového uživatele. Zkontrolujte, zda ve správci zařízení není mezi grafickými adaptéry SVGA 3D (Microsoft Corporation - WDDM, pokud ano, odinstalujte jej. Do počítače nainstalujte požadovaný software a upravte jej dle vašich požadavků pro virtuální desktopovou infrastrukturu. Po odhlášení ze vzdáleného připojení k VM stiskněte tlačítko View pro zobrazení důležitých věcí, které je nutno udělat.

Mezi těmito věcmi je:

Potvrzení správnosti license

Povolení vzdálené plochy pro uživatele domény

Povolení portů 1494, 2598, 3389 a 8080

Vypnutí automatických aktualizací, spořičů obrazovky, antivirů

Zaktualizovaný systém

Odinstalovaná grafická karta VMware SVGA 3D (MS WDDM)

Přidaný USB řadič pomocí vSphere Client.

Pokud jste obraz připravovali tak, jak jsem naváděl, máte vše v pořádku a můžete potvrdit každý řádek hodnotou Yes. Pokud ne, ještě je čas to napravit a poté vše potvrdit. Na další straně vybereme možnost Pooled and personal desktops, aby mohl mít každý uživatel v počítači svá data a potvrdíme tlačítkem Prepare. Až VDI-in-a-box dokončí přípravu obrazu, zkuste se do počítače přihlásit a překontrolujte, zda vše pracuje tak, jak požadujete. Pokud ne, vraťte se zpět na stránku s editací obrazu a VM opravte, v opačném případě obraz uložte. Takovýchto obrazů si později budete moci vytvořit více z nabídky images a každý takovýto obraz může obsahovat jiný software, dle skupiny, pro jakou bude určen.

Nyní si vytvoříme šablonu, pro ukázkou to mohou být třeba žáci. Jméno šablony tedy zadám Žáci, vyberu požadovaný obraz a zadám jednoduchý popis. Jméno virtuálního počítače se bude skládat z prefixu a suffixu. Prefix bude pevný, suffix bude pořadové číslo počítače a bude obsahovat tolik cifer, kolik jich zde zadáme. Když bude prefix zaci a suffix 000, názvy virtuálních počítačů budou vypadat takto: zaci000, zaci001,.. zaci039 apod. Pokud nelze zadat cifry na české klávesnici, zkuste přepnout na anglickou.

Když budeme chtít zpřístupnit ve virtualizovaném desktopu hardware připojený k fyzickému počítači (počítač, na kterém se připojujeme k VDI - u kterého sedí uživatel virtualizovaného počítače), můžeme to povolit zaškrtnutím některých z následujících položek. Například budeme chtít, aby si uživatel mohl připojit k PC USB zařízení, zaškrtneme možnost Other USB devices. Na další straně vybereme maximální počet desktopů na serveru spuštěných z této šablony a počet desktopů předem spuštěných. Můžeme si vybrat, zda budeme chtít desktopy personalizovat či nikoli - možnosti Pooled desktop a Personal desktop. V první možnosti nebude mít uživatel možnost uchovávat dlouhodobě svá data, v druhé bude každému uživateli přidělen disk o vybrané kapacitě.

K funkční virtuální desktopové infrastruktuře už zbývá jen vybrat uživatele, kteří se budou moci připojovat k VDI a přiřadit jim šablonu, ze které bude jejich desktop. Pokud máme v doméně AD např. skupinu zaci a těm budeme chtít zpřístupnit VDI, na záložce Users klikneme na odkaz Add vedle Users Groups a

tam napíšeme název skupiny a vybereme šablonu, která má být použita. Stejným způsobem můžeme přidávat jednotlivé uživatele o několik řádek níže.

Na záložce Desktops se přesvědčíme, že se desktopy startují a až bude připraven nějaký desktop k připojení (sloupec New), zkusíme se k němu připojit jako uživatel přes prohlížeč, do kterého zadáme adresu VDI-in-a-box a vyplníme uživatelské jméno a heslo doménového uživatele.

1.3.2 Klientský počítač

Server pro VDI je nastaven, ještě je potřeba připravit počítače, kterými se bude k virtualizovaným desktopům přistupovat. Jak jsem již řekl, není potřeba žádný výkonný počítač, virtualizovaný desktop zaměstnává hardware serveru. Je téměř jedno, jaký operační systém bude v tomto klientovi, záleží pouze na tom, aby bylo možné do něho nainstalovat Firefox(Iceweasel) nebo Operu a CITRIX Receiver. Já jsem počítač připravil takto: Nainstaloval jsem Debian 6 s grafickým rozhraním KDE a rozšířením kiosktool, Iceweasel a CITRIX Receiver. Kiosktool jsem nainstaloval kvůli ochraně počítače (lze počítač nastavit, aby byl jako internetový kiosek, aby se do něho nikdo "nedostal"), pro mnohé situace je to možná zbytečné. Do Iceweaselu jsem nastavil jako domovskou adresu VDI-in-a-box a zvolil jsem možnost, aby se při každém spuštění prohlížeče načítala tato adresa. Skryl jsem všechny lišty a při připojení k VDI-in-a-box trvale přijmul certifikát. Iceweasel se spouští automaticky po startu počítače.

V takto připraveném počítači se již stačí přihlásit pod uživatelským loginem a virtualizovaný desktop je možné používat.

Možností je skutečně mnoho, přistoupit k VDI je možné i z linuxu bez grafického rozhraní - do holého Debianu nainstalovat xserver, CITRIX receiver a prohlížeč internetu. Případně je možné k VDI přistupovat tabletem či telefonem - do zařízení s OS Android nainstalujte aplikaci Citrix Receiver a do ní zadejte adresu VDI-in-a-box serveru a uživatelský login.